



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN

Titulación:

INGENIERO INDUSTRIAL

Título del proyecto:

ANÁLISIS DEL IMPACTO DEL DIMENSIONAMIENTO DE LAS INSTALACIONES EN LA CERTIFICACIÓN ENERGÉTICA DE EDIFICIOS

Ion Irañeta López de Dicastillo

Miguel Ángel Pascual Buisán

Pamplona, 26 de Noviembre de 2010



ÍNDICE

1. INTRODUCCIÓN.....	4
1.1. CERTIFICACIÓN ENERGÉTICA DE EDIFICIOS	4
1.2. MIYABI	13
1.3. ¿CÓMO SURGE CE ³ X?	14
1.4. ¿QUÉ ES CE ³ X?	15
1.5. ¿QUÉ ES UN PLUGIN?	20
1.5.1. Dimensionamiento de las instalaciones en la certificación de edificios	21
2. ANTECEDENTES.....	22
2.1. NORMATIVA	23
3. OBJETIVOS.....	25
3.1. ARQUITECTURA BIOCLIMÁTICA	25
3.1.1. Definición	25
3.1.2. Tipos de edificación bioclimática	27
3.1.3. Conceptos y principios básicos	27
4. RECOPIACIÓN Y PRESENTACIÓN DE DATOS.....	37
4.1. DATOS E INFORMACIÓN NECESARIA PARA LA REALIZACIÓN DEL PLUGIN	37
4.2. DIFICULTADES EN LA IMPLANTACIÓN DEL MÉTODO	39
5. DESARROLLO DEL TRABAJO.....	42
5.1. CÓDIGO FUENTE DE LOS FICHEROS PROGRAMADOS	42
5.1.1. PluginsDES.py	43
5.1.2. ASHRAE.py	46
5.1.3. ASHRAETerciario.py	57
5.1.4. TablasASHRAE.py	77
6. ANÁLISIS DE LOS RESULTADOS.....	103
6.1. PluginsDES.py	103
6.2. ASHRAE.py	103
6.3. ASHRAETerciario.py	104
6.4. TablasASHRAE.py	108
7. CONCLUSIONES.....	111
8. MANUAL DE USUARIO.....	113
9. MANUAL DE PROGRAMADOR.....	117
9.1. LENGUAJE DE PROGRAMACIÓN	117
9.2. PYTHON	120
9.3. ENTORNO DE DESARROLLO INTEGRADO	125
10. BIBLIOGRAFÍA.....	129
10.1. LIBROS	129
10.2. NORMAS	129
10.3. PÁGINAS WEB	129
11. ANEXOS.....	130

1. INTRODUCCIÓN

1.1. CERTIFICACIÓN ENERGÉTICA DE EDIFICIOS

La certificación energética de edificios es un requisito legal que tienen que cumplir todos los edificios nuevos, y que próximamente también afectará a los edificios existentes. Casi el 40% del consumo de energía primaria es debido a los edificios, y por ello las normativas europeas han intentado incidir sobre el consumo energético de las construcciones, en este caso creando una herramienta similar a la ya empleada en el caso de los electrodomésticos.

El decreto obliga a clasificar las nuevas construcciones con una etiqueta que informe a los compradores del grado de eficiencia del edificio. Se trata de que cada edificio disponga de una etiqueta con su calificación energética (de la A, que correspondería a los edificios más eficientes, a la G, los edificios menos eficientes) y en la que se incluya su consumo estimado de energía y las emisiones de CO₂ asociadas.

Calificación Energética de Edificios proyecto/edificio terminado	
<div> <div>Más</div> <div> <div>A</div> <div>B</div> <div>C</div> <div>D</div> <div>E</div> <div>F</div> <div>G</div> </div> <div>Menos</div> </div>	
Edificio:	_____
Localidad/Zona climática:	_____
Uso del Edificio:	_____
Consumo Energía Anual:	_____ kWh/año (_____ kWh/m ² /año)
Emisiones de CO ₂ Anual:	_____ kgCO ₂ /año (_____ kgCO ₂ /m ² /año)
<p>El Consumo de Energía y sus Emisiones de Dióxido de Carbono son las obtenidas por el Programa _____ para unas condiciones normales de funcionamiento y ocupación.</p> <p>El Consumo real de Energía del Edificio y sus Emisiones de Dióxido de Carbono dependerán de las condiciones de operación y funcionamiento del edificio y de las condiciones climáticas, entre otros factores.</p>	

La etiqueta energética que deben mostrar todos los edificios debe incluir su clase de eficiencia y su consumo anual estimado. Aunque los datos son relativos, aportan al comprador algo más de información para decidir. Hay que recordar, sin embargo, que

cuando comparamos edificios muy diferentes entre sí, no siempre el de clase más alta consume menos y que en el consumo final los hábitos personales juegan un papel fundamental.

Sin embargo, los electrodomésticos se someten a un test en un laboratorio, mientras que los edificios son sistemas mucho más complejos cuyo funcionamiento no se puede testear o estimar de manera tan sencilla, sometidos a condiciones y hábitos de uso mucho más variables. Por ello, poner en práctica un sistema de este tipo no es posible si no es realizando grandes simplificaciones. Además, la metodología establecida para la certificación tiene diversos puntos controvertidos, como se expondrá más adelante, y todavía hay un elevado grado de confusión e incertidumbre en el sector, que da una idea de la baja implementación voluntaria de la certificación de edificios.

El objetivo de la certificación de edificios sería incentivar a los promotores a construir edificios más eficientes y animar a la rehabilitación de edificios para que consumieran menos energía. Esto se conseguiría porque, en primer lugar, una promoción con una calificación más eficiente tendría una mejor imagen, sumaría otro argumento para su venta y, en segundo lugar, la existencia de un etiquetaje facilitaría que el consumo de energía se convirtiese en un criterio más de compra por parte del consumidor.

Detalles de la certificación energética

La Directiva Europea 2002/91/CE tiene como objetivo fomentar la eficiencia energética de los edificios y obliga a todos los estados miembros, entre otras cosas, a que todo edificio, tanto si se vende como si se alquila, vaya acompañado de un certificado de eficiencia energética. Este certificado se presentaría a la persona interesada, el propietario o inquilino. Esta directiva en el estado español no se ha transpuesto hasta el año 2007, mediante la aprobación del Código Técnico de la Edificación (CTE), la realización de modificaciones al Reglamento de Instalaciones Térmicas de edificios (RITE), y el Real Decreto 47/2007, de 19 de enero, que define la aplicación de una certificación energética.

Este decreto entró en vigor el 30 de abril de 2007, fue de aplicación voluntaria hasta el 31 de octubre de 2007, mientras que a partir de entonces su cumplimiento es obligatorio.

Así, la certificación energética de edificios, por ahora, aplica a todas las nuevas construcciones y a las grandes rehabilitaciones (se consideran grandes rehabilitaciones las de más de 1000 metros de superficie y en las que se renueve más del 15 % de los cerramientos). Las excepciones que están exentas de aplicar la certificación son:

- Construcciones provisionales con un plazo de utilización igual o inferior a 2 años.
- Edificios industriales o agrícolas.

- Edificios aislados con una superficie útil total inferior a 50 m².
- Edificios de sencillez técnica y escasa entidad constructiva que no tengan carácter residencial o público (desarrollados en una sola planta y que no afectan a la seguridad de las personas).
- Edificaciones que por sus características de utilización deban permanecer abiertas.
- Edificios y monumentos protegidos oficialmente, cuando el cumplimiento de este decreto pudiese alterar de manera inaceptable su carácter o aspecto.
- Edificios utilizados como lugares de culto o para actividades religiosas.

El decreto que recoge la certificación energética resulta bastante general. Define la obligatoriedad de la certificación y los requisitos de los programas informáticos que se deben emplear, pero deja en manos de las comunidades autónomas el desarrollo de procedimientos de implantación y, muy importante, el control de esta certificación energética. Así, son las autonomías las que, a partir de este decreto o directamente desde la directiva si tienen competencias para ello, deben establecer los procedimientos administrativos necesarios, el alcance y características de los controles al edificio para garantizar la veracidad del certificado y otros temas, como el procedimiento para la renovación del certificado.

De todas formas, se debería trabajar tanto en el procedimiento en sí, como en la formación de los agentes de la construcción y también en la información a la población en general. Esto es un punto fundamental si realmente se quiere que la certificación no sea tan sólo una normativa a cumplir, sino un detonante que impulse la eficiencia en los edificios. Si el público no está sensibilizado para elegir, el sistema no tiene sentido.

En resumen, la certificación energética quiere ser una evaluación cuantitativa y objetiva del comportamiento energético del edificio, que debe ser presentada de forma comprensible al usuario. Para realizar esa evaluación del edificio, se ha establecido una metodología de cálculo, y para hacer llegar los resultados al usuario, una etiqueta tipo. A continuación se describen estos dos componentes de la certificación.

Metodología de cálculo

Para poder llegar a calificar energéticamente el edificio se realiza una modelización teórica del consumo energético del edificio. Esto es así porque el certificado y la clase de eficiencia deben estar disponibles cuando el edificio se vaya a vender, no cuando ya esté siendo utilizado. Además, tan sólo se podría tratar de comparar edificios en unas condiciones teóricas, ya que cuando estuviera habitado el consumo energético influiría según los hábitos de cada casa.

Así, en la fase de proyecto se debe calcular la demanda de energía del edificio a proyectar. Para este cálculo de la eficiencia del edificio el único programa reconocido actualmente es el programa **LIDER**. El programa modelizaría los datos de consumo eléctrico total, a partir de una descripción del edificio que incluiría características de la envolvente, la ventilación y orientación, las condiciones ambientales interiores, la existencia de sistemas solares pasivos y protecciones solares, las instalaciones de calefacción, ACS y aire acondicionado y las de iluminación.

A continuación se pasaría a calcular la calificación energética del edificio, es decir, la eficiencia energética de un edificio respecto a uno convencional. Hasta el momento, el único programa reconocido es el **CALENER**. Ambos programas están accesibles públicamente en el registro de documentos reconocidos del Ministerio de Industria, Turismo y Comercio. La herramienta CALENER compararía el edificio modelado con un edificio “estándar” de características similares, situado en la misma localidad geográfica y cuyo comportamiento energético ha sido analizado en un estudio de campo. En función de esa comparación, el programa le asigna una clase de eficiencia energética, de la A a la G.

Finalmente, con esta calificación en una categoría (de la G a la A) se emiten el certificado energético y la etiqueta provisionales. Posteriormente, durante la ejecución del edificio, mientras está siendo construido y acabado, se comprobaría que esta eficiencia simulada en la fase proyecto coincide con el funcionamiento energético real. Se obtiene la calificación energética del edificio acabado, se reajustan los datos a la calificación adecuada si es necesario, y se otorga el certificado definitivo.

El certificado de eficiencia energética se debe incluir en el libro del edificio y tiene una vigencia de 10 años, tras los cuales el edificio se debería volver a calificar para que la etiqueta que muestre sea adecuada a la realidad del edificio.

La emisión y registro de certificados y los procedimientos de verificación y control son competencia de las comunidades autónomas y todavía están por definir, aunque ya se puede prever la inmensa dificultad que supondrá realizar el seguimiento.

Opciones de cálculo para la aplicación práctica de la certificación de edificios

Los edificios son sistemas complejos, e intentar modelizarlos y compararlos es extremadamente difícil sin caer en simplificaciones. Por ello, a la hora de obtener la calificación del edificio, se ha propuesto poder escoger entre dos opciones (general y simplificada), y según la opción escogida se puede acceder a una clase de eficiencia u otra. Ello podría restar sentido y fuerza a la etiqueta, como se verá.

La **opción general** permite acceder a las calificaciones superiores (clases A, B, C). En

ella se utilizan los programas de cálculo de referencia, como se ha dicho, que deben estar reconocidos como tales para ser válidos para obtener el certificado, y están disponibles públicamente. Actualmente son los ya nombrados LIDER y CALENER.

Dentro de esta opción, se ha realizado una distinción entre los edificios destinados a viviendas y el resto, ya que en los primeros se puede esperar un comportamiento más homogéneo que en los segundos, que incluyen usos más diversos como hospitales, edificios agrícolas, locales comerciales, etc.

Así, para los edificios destinados a viviendas, se hizo un estudio de campo seleccionando 14 edificios de viviendas unifamiliares y 11 plurifamiliares, con las 4 orientaciones posibles y en las 12 localizaciones climáticas posibles según el CTE. Con estas características, se calcularon las emisiones totales asociadas a cada una de esas tipologías. Entonces, la calificación general para viviendas se obtendría por comparación con uno de estos edificios de referencia, el más similar en tipología al edificio que deseamos calificar.

Por otro lado, para los edificios no residenciales, en los que sería muy difícil generar unas tipologías estándar con sus correspondientes consumos y emisiones estimados, la comparación se realiza con un edificio ficticio, llamado edificio de referencia, que se caracteriza por tener el mismo diseño que el edificio a calificar y cumplir los requisitos mínimos que marca el CTE.

La **opción simplificada** se basa en el cumplimiento de los mínimos que marca el CTE. Si se utiliza esta metodología más sencilla para el cálculo de la eficiencia del edificio, sólo se pueden obtener las clases D o E. Por ello, un edificio podría obtener una calificación más baja si en el cálculo se utiliza la opción simplificada que si se utiliza la opción general. La clase de eficiencia en este caso se obtiene de manera indirecta, a través de soluciones técnicas que mejorarían el comportamiento energético del edificio respecto a los requisitos básicos que marca el CTE. Estas soluciones técnicas están recogidas en una guía, y la adopción de unas u otras determinan la calificación definitiva, D o E. Las soluciones sólo incluyen un parámetro de compactidad y las eficiencias de las máquinas de calefacción, refrigeración y ACS, y además, en algunas de estas categorías no hay definidos parámetros, o todas las eficiencias son permitidas, por lo que no hay un gran margen para diferenciar o premiar soluciones eficientes. En definitiva, esta opción, que busca facilitar algo el proceso, haría perder parte del valor de comunicación de la etiqueta, ya que no podríamos comparar edificios que hayan sido calificados con opciones diferentes (y en la etiqueta final el método de calificación que se ha seguido no aparece, por lo que esta información no llega al consumidor). También podría hacer temer una avalancha de edificios de clase D o E, que quizá hubieran podido obtener una calificación mejor, debido a la mayor facilidad del procedimiento y a la falta de información de los consumidores: si no se hace

previamente el trabajo de información y sensibilización que el tema requiere, el público final no demandará edificios más eficientes.

Etiqueta y clases de eficiencia energética

La información al consumidor se debe facilitar en forma de una etiqueta que se otorga al edificio de manera provisional y después definitiva. Esta etiqueta está normalizada y es estéticamente similar a la que ya se aplica a los aparatos electrodomésticos, con un código de letras (**A para edificios más eficientes, G para edificios menos eficientes**) y colores (verde más eficiente, rojo menos). También como en el caso de los electrodomésticos, las diferencias entre clases de eficiencia energética serían importantes. Por ejemplo, en la aplicación de la calificación sobre un proyecto de dos viviendas unifamiliares situadas en Madrid, se estimó que la calificada como clase de eficiencia A emitiría 4 veces menos CO₂ que la calificada como E.

En todo caso, habría que tener en cuenta que *sólo podremos comparar la clase de eficiencia energética de dos edificios cuando sean muy similares entre sí*. Esto es así porque todos los edificios se califican *en relación a un edificio similar a él*, con un comportamiento energético determinado, pero **no siempre el edificio más eficiente consumirá menos**.

Por ejemplo, un edificio de clase A puede consumir más que uno de clase C pero que es mucho más pequeño, aunque su peor aislamiento le hace tener una calificación más baja.

Así, podremos decidir según la clase de eficiencia, cuando comparemos edificios de características similares. En otros casos, además, podremos fijarnos en el dato de consumo de energía previsto.

Así, otras informaciones que aparecen en la etiqueta son la localidad y zona climática en la que se encuentra el edificio (de acuerdo con la sección HE1 del CTE), el uso que tiene (residencial, comercial...) y el valor numérico del **consumo de energía primaria** estimado del edificio, expresado en **kWh/año**, las emisiones correspondientes de dióxido de carbono (**kg CO₂/año**), así como los mismos datos por m² de superficie.

Cuando nos fijemos en los valores de consumo previsto sucederá lo mismo que en el caso de los electrodomésticos: una nevera americana de clase A + consumirá mucho más que una pequeña nevera con congelador pequeño de clase B. Así, hay que considerar también las prestaciones que queremos y el consumo previsto (kWh/año) además de la clase.

Es de destacar que, como todo sistema de etiquetaje ecológico, el sistema de certificación no tiene sentido si esta etiqueta no es reconocida más allá de, en este caso,

el sector de la construcción, que se ve obligado a implementarlo. En función de la divulgación que se haga, y más allá de las limitaciones que presenta el procedimiento de certificación, se logrará o no que el público en general comience a incluir conscientemente criterios medioambientales en la que seguro es la compra más importante de su vida. No es de recibo insistir en la necesidad de difusión del sistema y la etiqueta si tenemos en cuenta que, tras más de 15 años de funcionamiento, el etiquetaje de electrodomésticos todavía sigue siendo desconocido por mucha gente.

Comentarios al sistema de certificación energética

En resumen, la certificación energética, la miremos como un mero requisito legal, o como un proceso de promoción de una construcción más respetuosa, merece unas cuantas reflexiones o que tengamos algunas cuestiones en cuenta:

- Las recientes normativas concernientes a la edificación (Código Técnico, certificación energética...) todavía, incluso estando ya en momento de aplicarlas, se perciben por parte de los agentes principales (arquitectos, promotores...) como normas de difícil aplicación y más difícil comprobación. En general, la percepción es que se trata simplemente de una nueva imposición, y no un sistema en el que la mejora energética de los edificios sea un aliciente, o que pueda motivarles a diferenciarse con promociones más ecológicas. Es difícil predecir cómo llegará la información energética del edificio al usuario final si los propios agentes que deben comunicarla a sus compradores, publicitarla, etc., no las tienen todas consigo.
- Los edificios se comparan con las posibilidades de ahorro o consumo de energía que podría tener en el lugar en el que se encuentra, y según su tipología. Por ello, la clase de eficiencia no es un criterio absoluto de elección, sino que tendremos que valorar, como en el caso de los electrodomésticos, no sólo la clase de eficiencia sino los valores de consumo. Decidir por la clase de eficiencia será útil cuando comparemos edificios muy similares en superficie y prestaciones.
- Hay que tener en cuenta que la existencia de la opción simplificada hace que la calificación de un edificio, la letra que lo define, dependa también de qué sistema ha usado. Una vez calificado el edificio, haber utilizado la opción general no tiene más valor que haber escogido la simplificada. Además, resulta más difícil conseguir calificaciones altas, porque en la opción general se debe modelar el edificio con las complicadas herramientas informáticas existentes. Estas herramientas, además, podrían vetar algunos diseños que podrían suponer ahorros energéticos importantes, simplemente porque su introducción y demostración a través de esos programas informáticos sería muy complicada. Esto resta algo de sentido al propio etiquetaje, ya que entonces una etiqueta que debería servir para comparar de manera universal no permite la comparación. Por ello, de nuevo sería importante fijarse, además de la

calificación, en el dato de consumo, para el que todos los edificios habrían utilizado el mismo programa.

- Sin embargo, aquí aparece otro problema. Hemos utilizado el condicional al hablar de que “sería” importante fijarse en el consumo estimado del edificio. Efectivamente, podría ser un parámetro importante de decisión, pero este número se debe obtener con el programa LIDER, que tiene numerosas limitaciones.

- Las herramientas informáticas que se deben emplear para hacer los cálculos son complejas y controvertidas. Sólo existen como sistemas reconocidos oficialmente el LIDER y CALENER. Respecto a usabilidad, son muy cerrados y poco prácticos y no admiten soluciones constructivas que se ha comprobado que podrían incidir y mucho sobre el consumo de energía de los edificios, como por ejemplo los invernaderos. También presentan muchas limitaciones en cuanto a la modelización teórica que harían del comportamiento térmico del edificio con la irradiación solar, o las potencias de los sistemas, cuestiones que finalmente determinan el consumo real de una vivienda. De hecho el problema básico es que estos programas se han planteado para verificar el cumplimiento de la normativa, del CTE, y por tanto no permiten o son poco efectivos para premiar soluciones más innovadoras que vayan más allá de las normativas en cuanto a sostenibilidad.

- Todavía está por definir la comprobación de la calificación, la aplicación a los edificios existentes y la metodología para la renovación de certificados, al cabo de los 10 años de vigencia. Aunque se trata de propósitos admirables, dada la situación de retraso actual, las dificultades de realizar un control y las limitaciones de los datos que aporta la etiqueta, el sistema corre el peligro de quedar como una normativa de resultados descafeinados.

- Como ya se ha destacado varias veces, la existencia de un etiquetaje en los edificios sólo servirá para crear edificios más eficientes si, primero, existe suficiente información sobre criterios medioambientales en el sector y voluntad de aplicarlos y, segundo, si el público en general está concienciado y exige calificaciones más altas. Tanto una cosa como la otra aún no se dan, y peor aún, el tema de la vivienda es un tema delicado y la sensibilización en este tema se enfoca a los problemas de acceso, etc. Por ahora, en general las preocupaciones de la población en cuanto a la vivienda no son, precisamente, sus impactos ambientales.

- La certificación energética se plantea dentro de una actualización de la normativa relativa a la edificación que presenta algunos problemas de base para poder cumplir sus objetivos. Así, el CTE obliga a incluir en el proyecto parámetros difíciles de controlar posteriormente, en vez de plantear el sistema a la inversa. No limita el consumo de energía (a diferencia de los sistemas de otros países como Alemania), sino


que se ha seguido en la línea de la normativa anterior, que lo único que determinaba eran limitaciones al coeficiente global de transferencia del edificio. Si la limitación de la demanda energética se planteara a partir de un objetivo final, por ejemplo, de un consumo energético bajo, de menos de x kWh por año y metro cuadrado, este objetivo se podría lograr a través de las más variadas y diversas soluciones constructivas. Con el sistema planteado actualmente, hay que ceñirse a soluciones estándar y limitaciones en el diseño constructivo para formar parte de los patrones aceptados por unos rígidos programas informáticos. Un ejemplo recogido en un estudio sobre el tema es la realidad al aplicar la normativa a una casa bioclimática: en primer lugar, estaba fuera de la opción simplificada porque disponía de más de un 60 % de acristalamiento, y en segundo lugar, la opción general prestacional no la podía evaluar bien porque el programa no permitía la inclusión de elementos como suelo radiante, sombreado activo, etc. y encima la compararía con un edificio de referencia con menos ventanas y más superficie aislada. El comportamiento energético que te permite simular el programa es lo que determina que se acepte o no, y se pueden quedar fuera de norma, o con una calificación peor, edificios que en la práctica tendrían mejor comportamiento energético.

- Si se planteara un límite de consumo establecido, que incluso incluyera el uso por parte del ocupante final del edificio, la comprobación sería extremadamente sencilla (una simple monitorización de consumos) y seguro que se motivaría la adecuada información a los usuarios sobre los sistemas de que disponen y como utilizarlos para el ahorro de energía. Parecería una idea descabellada, y obviamente, entrar por la vía impositiva no haría más que crear incomodidad. Pero también es cierto que este tipo de sistemas de certificación si funcionan finalmente es gracias a la voluntad de que funcionen, dado que las medidas de comprobación son más bien débiles. Por ello, quizá la solución no sería crear complejas normativas que *obliguen* a la realización de unos cuantos papeles más, sino de crear sistemas en los que se *motive* construir mejor y por iniciativa propia interese mostrar lo que se ha hecho. Y para ello, hay que insistir una y otra vez en la necesidad de la información a la población.

- Finalmente, por la propia concepción del sistema se analiza solamente el consumo energético del edificio durante su funcionamiento. Entendemos que la directiva trata exclusivamente del comportamiento energético, y que el principal reto ecológico al que se enfrenta la sociedad actualmente sería el del cambio climático, directamente relacionado con el uso desmesurado de combustibles fósiles para obtener energía. Sin embargo, gran parte de la energía de un edificio es la utilizada para manufacturar los materiales que lo forman y transportarlos al sitio. Además, la energía no es el único impacto ambiental de un edificio. Y si se comenzasen a incorporar legislaciones o certificaciones para cada uno de los vectores con impactos ambientales, la complejidad y el trabajo de adaptarse a todas ellas (agua, reciclabilidad y toxicidad de materiales...) sí que se podría convertir en una carga de trabajo inasumible.

En resumen, la nueva normativa y el sistema de certificación introducen nuevos requisitos respecto la sostenibilidad en los edificios, pero no se crea un procedimiento que realmente motive construir más ecológico. La legislación se ha planteado una vez más como una imposición burocrática. Aunque los resultados están por llegar, no parece que estas novedades vayan a acelerar demasiado la necesidad urgente de que los parámetros ambientales se incorporen en el candente sector de la construcción, si todavía no los incorporamos en las pequeñas cosas de cada día. Hoy por hoy, la sostenibilidad en los edificios avanza poco a poco, y de momento el consumo final de los edificios depende, más allá de normativas, del experto que proyecta y construye, de la persona que compra o rehabilita un piso, o del usuario que elige actuar de una u otra manera, en definitiva, de una cuestión de conciencia ambiental y voluntad personal.

1.2. MIYABI

MIYABI -eficiencia energética en la edificación- es una sociedad pública del gobierno de Navarra. 

Fundada en el año 2001, el objeto social de la empresa es la promoción de eficiencia energética, uso racional de la energía, e integración de renovables, en viviendas y espacios urbanos bioclimáticos diseñados con el fin de conseguir un sistema energético sostenible, manteniendo al mismo tiempo el equilibrio de los ecosistemas y fomentando el desarrollo económico.

Las principales **actividades** son:

1.- Consultoría y asesoramiento en eficiencia energética y sostenibilidad.

Desde la concepción del edificio, hasta su supervisión una vez terminado,

MIYABI realiza las siguientes actividades:

- Análisis de alternativas constructivas
- Cálculo de amortizaciones energéticas
- Análisis de ciclo de vida
- Calificación energética de edificios (CALENER VYP, CALENER GT)
- Estudio de idoneidad de sistemas
- Análisis de confort térmico

2.- Promoción de vivienda y terciario

3.- Realización de estudios de investigación y desarrollo:

Desarrollo de metodologías y aplicaciones informáticas para el análisis energético:

- Certificación Energética de Edificios Nuevos: CES
- Certificación Energética de Edificios Existentes: CE³X

1.3. ¿CÓMO SURGE CE³X?

Como se ha comentado, MIYABI está íntimamente ligado al mundo de la eficiencia energética y aplica sus conocimientos en el ámbito de la vivienda. Además de los edificios con uso residencial, da cobertura también a los edificios para uso terciario, ya sean pequeños edificios (Pequeño Terciario) o grandes edificios (Gran Terciario). Y todo esto gracias a dos herramientas informáticas desarrolladas que diferencian los edificios nuevos de los existentes.

El consorcio formado por CENER y MIYABI, que cuenta también con la participación de CENIFER (Centro Nacional Integrado de Formación en Energías Renovables), resultó adjudicatario en el concurso público para desarrollar procedimientos de certificación energética de edificios en España.

El concurso público fue convocado por el Ministerio de Industria, a través del Instituto para la Diversificación y Ahorro de la Energía (IDAE), y a él concurrieron 17 agrupaciones formadas por empresas, centros tecnológicos y universidades de España. El IDAE adjudicó este concurso al consorcio formado por CENER y MIYABI, organismos que están diseñando el método y el programa informático que se utilizarán en la certificación energética de bloques de viviendas, oficinas y servicios ya construidos en España.

Como ya es sabido, a partir de 2007, todos los edificios de nueva construcción deben incorporar en España medidas de ahorro energético y una certificación energética. Además, el Gobierno, en cumplimiento de las directivas comunitarias en la materia, está ultimando la normativa que obliga a los edificios ya existentes a contar también con una certificación energética. Esta normativa establecerá, por ejemplo, la obligación de contar con una certificación si se quiere vender o alquilar una vivienda.

La adjudicación a CENER y MIYABI de los procedimientos de certificación tiene gran trascendencia social, económica y ambiental, por diferentes aspectos:

1. El alcance de la adjudicación, que abarca a los pisos, oficinas, comercios u hospitales de todo el país. (De acuerdo con datos del Banco de España, sólo el parque de pisos existente en España alcanza los 25,5 millones de viviendas).
2. El sector residencial y servicios supone más del 30 % del gasto de consumo de energía final en España (fuente: IDAE). De ahí la importancia de incidir en las medidas de uso eficiente de la energía en ese ámbito.

3. La aplicación de medidas de ahorro en los edificios supondrá un elemento de generación de actividad económica y empleo en diferentes sectores. (Los certificadores de edificios serán formados mediante un sistema que implante CENIFER).
4. Esta adjudicación refuerza la posición de liderazgo de Navarra en materia de energías renovables y ahorro.

La herramienta diseñada por las entidades adjudicatarias consiste en un software práctico, rápido y sencillo que permite obtener la calificación energética del edificio y realizar un estudio de las medidas de ahorro energético más eficientes en cada caso. Los profesionales del sector podrán, de forma inmediata, realizar un estudio de las medidas más eficientes sobre el edificio objeto para mejorar su calificación y disminuir sus emisiones de CO₂. Así, la aplicación se vuelve una guía dinámica de recomendaciones concretas para el edificio objeto.

El Centro Nacional de Energías Renovables de España (CENER) es un centro tecnológico especializado en la investigación aplicada y el desarrollo y fomento de energías renovables. CENER cuenta con más de 200 investigadores y tiene actividad en los cinco continentes. El Patronato de CENER está compuesto por el Ministerio de Ciencia e Innovación, Ciemat, el Ministerio de Industria y el Gobierno de Navarra. Desarrolla su actividad en seis áreas de trabajo en el campo de las energías eólica, solar térmica y solar fotovoltaica, biomasa, arquitectura bioclimática, e integración en red de energías renovables. Su sede está ubicada en la Ciudad de la Innovación (Sarriguren), aunque cuenta con oficinas en otros puntos de España. Dispone de modernos laboratorios acreditados e instalaciones a nivel europeo, como es el caso del Laboratorio de Ensayo de Aerogeneradores (ubicado en Sangüesa), un laboratorio de biomasa, un laboratorio de ensayo de paneles térmicos y módulos fotovoltaicos, así como un laboratorio de materiales y procesos de células fotovoltaicas.

1.4. ¿QUÉ ES CE³X?

CE³X es un programa de Certificación Energética de Edificios Existentes que se está desarrollando en la actualidad, ya en su fase final, y que será uno de los dos que operen en España y sea utilizado por los certificadores autorizados.

CE³X tiene dos características fundamentales que lo hacen atractivo en su conjunto. Una es su sencillez, ya que en cuanto se empieza a utilizar, ya se ve que su manejo es fácil, y otra es que es un programa bastante intuitivo.

La aplicación informática CE³X, para la obtención del certificado de eficiencia energética de edificios existentes muestra en la pantalla principal los siguientes componentes:

Barra de título

En la *barra de título* aparece el nombre de la aplicación (CE³X-res/CE³X-PT/CE³X-GT correspondiente a Certificación Energética de Edificios Existentes-Residencial/Pequeño terciario/Gran terciario) junto con el nombre del fichero que se encuentra abierto y su ubicación en el disco duro del ordenador.

Menú Principal

Es el elemento desde el que se accede a las diferentes funcionalidades del programa:

Archivo

- **Nuevo**, abre un nuevo archivo sin nombre el cual posee todos los campos libres para su cumplimentación.
- **Abrir**, permite abrir un archivo existente indicando la ruta y el nombre del archivo que se quiere abrir.
- **Guardar**, sobrescribe el archivo sobre el que se está trabajando, guardando bajo el mismo nombre y ruta las modificaciones realizadas desde la última grabación. En aquellos casos de archivos nuevos, en los que aún no se haya asignado un nombre ni una ubicación al archivo, el programa solicitará el nombre y la ruta del archivo a guardar.
- **Guardar como**, permite guardar el archivo generado con un nuevo nombre y en una nueva ruta.
- **Salir**, finaliza la aplicación.

Elementos constructivos

- **Librería de Materiales**, permite modificar o crear materiales que no estén definidos, en la librería existente en el programa, para su posterior utilización en la composición de los cerramientos del edificio a analizar.
- **Definir composición de cerramientos**, permite definir los distintos tipos de cerramientos que conforman el edificio en función de los distintos materiales que los componen.
- **Librería de vidrios/marcos**, permiten modificar o crear vidrios y marcos que no estén definidos en la librería de vidrios/marcos existente en el programa, para su posterior utilización en la composición de los huecos del edificio a analizar.
- **Librería de puentes térmicos**, permite modificar o crear puentes térmicos que no estén definidos en el programa para su posterior utilización en la definición de los puentes térmicos del edificio a analizar.
- **Obstáculos remotos**, permite definir (mediante proyección cilíndrica) los patrones de sombra que los obstáculos remotos proyectan sobre el edificio a analizar.

Resultados

- **Calificar**, procede a la calificación del edificio/vivienda en función de sus características, introducidas en el programa a través de los paneles de introducción de datos.
- **Generar informe**, permite generar e imprimir el informe con los resultados de eficiencia energética y valor de la certificación del edificio/vivienda así como el resultado de las propuestas de la mejora más significativas.
- **Medidas de mejora**, permite analizar medidas individuales y conjuntos de medidas de mejora aplicables al edificio/vivienda. Cada conjunto de medidas proporciona un nuevo valor de calificación y/o eficiencia energética del edificio analizado. Estas medidas de mejora se clasifican según su importancia en función de su coste y de la mejora producida en el resultado de la calificación.
- **Análisis económico**, realiza el análisis económico de las medidas de mejora propuestas en función del tiempo de amortización de la medida y del ahorro energético producido por la misma a lo largo de su vida útil.
- **Solicitar soporte técnico**, permite acceder a las últimas versiones del programa desde la página web oficial del CE³X y del Instituto para la Diversificación y Ahorro de la Energía (IDAE).

Ayuda

- **Consultar manual de usuario**, carga el manual de usuario con su índice que nos permite acceder de manera rápida al apartado el cual contiene dudas surgidas.
- **Consultar manual de casos prácticos**, carga los documentos de casos prácticos para su consulta.
- **Consultar guía de medidas de mejora**, carga la guía de medidas de mejora para su consulta.

Barra de herramientas

La barra de herramientas permite acceder a las funciones existentes en el menú principal pero de forma rápida y directa, como son: *abrir proyecto nuevo*, *cargar proyecto desde archivo*, *guardar*, *deshacer*, *definición de obstáculos remotos*, *calificar*, *medidas de mejora* *análisis económico* y *generar informe*.

Pestañas de entrada de datos

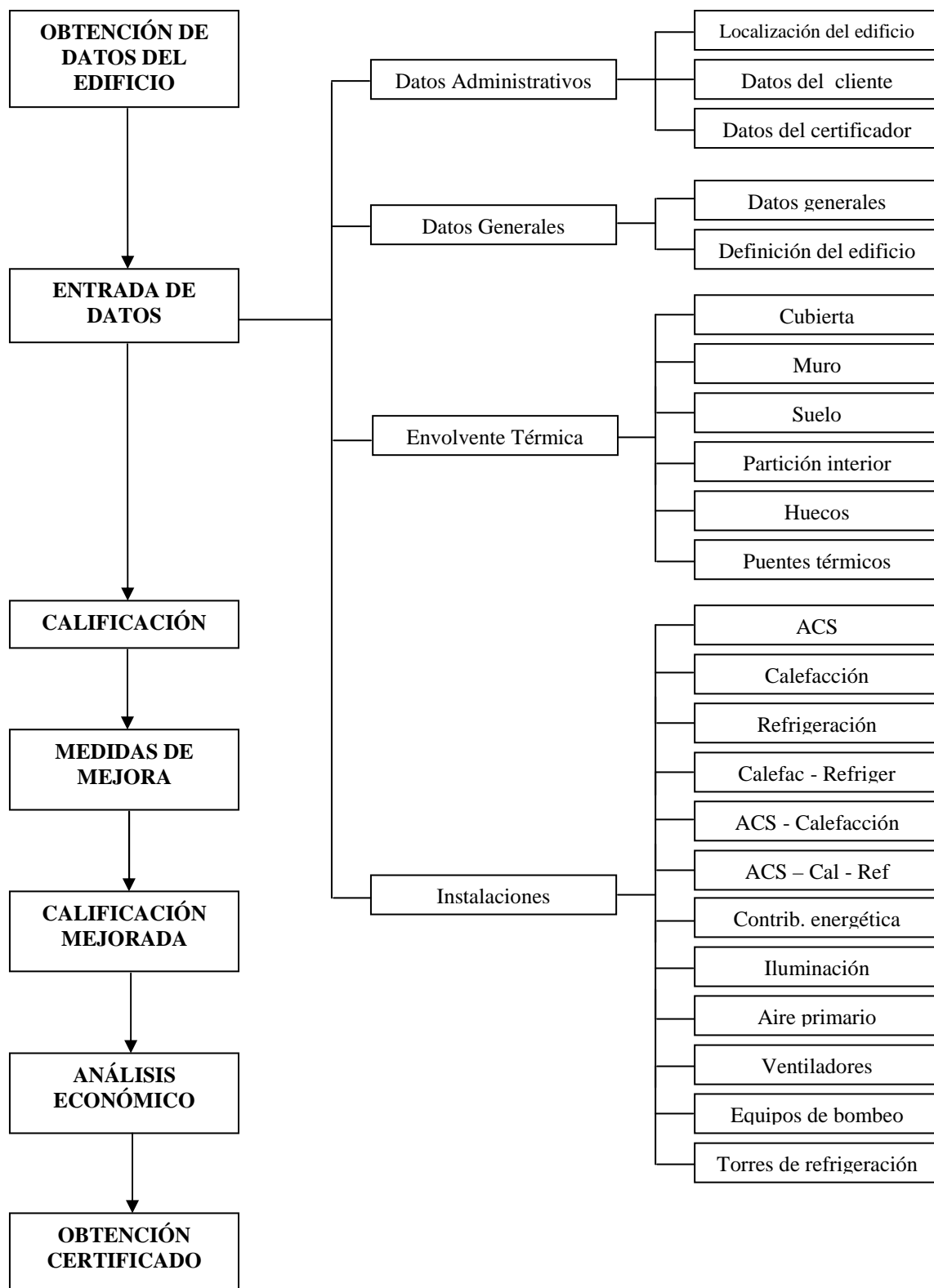
Se compone de una sucesión de paneles que permiten la introducción de todos los datos y características necesarias para definir la envolvente y la eficiencia de las instalaciones del edificio/vivienda, permitiendo la calificación energética, así como la introducción de las diferentes medidas de mejora y datos para el análisis económico de las mismas. Aquellas pestañas de entrada de datos que sirven para la descripción de las características del edificio/vivienda existente para su calificación se encuentran siempre visibles y son las siguientes:

- **Datos Administrativos.** Registra la información administrativa relativa al proyecto que no influye en el valor de la calificación. Son aquellos datos relativos a la *Localización del edificio o vivienda*, *Datos del Cliente* que encarga la certificación o *Datos del Certificador* que asume la responsabilidad de la misma.
- **Datos Generales.** Engloba información fundamental para la calificación del edificio.
- **Envolvente Térmica.** Desde esta pestaña se accede a los paneles de introducción de datos de los distintos elementos que componen la envolvente térmica que son: cubiertas, muros, suelos, particiones interiores en contacto con espacios no habitados, huecos/lucernarios y puentes térmicos.
- **Instalaciones.** Desde esta pestaña se accede a los paneles de introducción de datos de los sistemas de ACS, Calefacción, Refrigeración, Contribuciones energéticas y el los casos de edificio terciario Iluminación, Aire primario, Ventiladores, Sistemas de bombeo y Torres de refrigeración.

Las pestañas de resultados e introducción de datos referentes a las diferentes medidas de mejora aplicables al edificio y el análisis económico de dichas medidas de mejora se sumarán a estas pestañas iniciales según el certificador las vaya incorporando. Estas nuevas pestañas se incorporarán previamente a la emisión del informe final y serán:

- **Calificación Energética,** desde esta pestaña se obtiene el resultado de calificación energética en función de las características del edificio/vivienda existente.
- **Medidas de Mejora.** Permite aplicar modificaciones en la envolvente e instalaciones del edificio con el fin de mejorar el valor de su calificación final.
- **Análisis Económico.** Se introducirán en esta pestaña los datos referentes a: facturas energéticas del edificio/vivienda, definición de los datos económicos y valoración económica de las distintas medidas de mejora introducidas en el apartado anterior, obteniendo los resultados en periodos de amortización de las distintas medidas de mejora.

La introducción de datos al programa sigue las indicaciones del siguiente esquema:



1.5. ¿QUÉ ES UN PLUGIN?

Un **plug-in** (del inglés "enchufable"), es una aplicación que se relaciona con otro sistema más grande para aportarle una función nueva y generalmente muy específica. La idea es que el nuevo componente se enchufa simplemente al sistema existente. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de las interfaces de programación de aplicaciones (API). También se lo conoce como add-on (agregado), complemento, conector o extensión.

Los complementos permiten:

- que los desarrolladores externos colaboren con la aplicación principal extendiendo sus funciones
- reducir el tamaño de la aplicación
- separar el código fuente de la aplicación a causa de la incompatibilidad de las licencias de software

En general, cualquier aplicación puede añadir soporte para complementos.

Funcionamiento

La aplicación principal proporciona servicios que el complemento puede utilizar, incluyendo un método para que los complementos se registren a sí mismos y un protocolo para el intercambio de datos. Los complementos dependen de los servicios prestados por la aplicación de acogida y no suelen funcionar por sí mismos. Por el contrario, la aplicación principal funciona independientemente de ellos, lo que permite a los usuarios finales añadir y actualizar los complementos de forma dinámica sin necesidad de hacer cambios a la aplicación principal.

Las APIs proporcionan una interfaz estándar, lo que permite a terceros crear complementos que interactúan con la aplicación principal. Una API estable permite que complementos de terceros funcionen como la versión original y amplíen el ciclo de vida de las aplicaciones obsoletas.

Interfaz de programación de aplicaciones

Una interfaz de programación de aplicaciones o API (del inglés *application programming interface*) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usados generalmente en las bibliotecas.

Características

Una interfaz de programación representa una interfaz de comunicación entre componentes de software. Se trata del conjunto de llamadas a ciertas bibliotecas que ofrecen acceso a ciertos servicios desde los procesos y representa un método para conseguir abstracción en la programación, generalmente (aunque no necesariamente)

entre los niveles o capas inferiores y los superiores del software. Uno de los principales propósitos de una API consiste en proporcionar un conjunto de funciones de uso general, por ejemplo, para dibujar ventanas o iconos en la pantalla. De esta forma, los programadores se benefician de las ventajas de la API haciendo uso de su funcionalidad, evitándose el trabajo de programar todo desde el principio.

1.5.1. DIMENSIONAMIENTO DE LAS INSTALACIONES EN LA CERTIFICACIÓN DE EDIFICIOS

El presente Trabajo técnico tiene por objeto la realización de un plugin para el programa CE³X. Consiste en el diseño y programación de un plugin que sea capaz de dimensionar las instalaciones en el programa de certificación energética de edificios. Para ello, dicho complemento, calcula la potencia de calor que es necesario introducir en la vivienda para mantener unas determinadas condiciones de diseño, así como la potencia de refrigeración, para alcanzar las condiciones de diseño durante la época de verano.

Todos estos cálculos han sido desarrollados por un método de la ASHRAE, The American Society of Heating, Refrigerating and Air-Conditioning Engineers. La ASHRAE fue fundada en 1984 y se trata de una organización internacional formada por más de 50.000 personas. Su misión dentro del mercado de la ventilación, la calefacción y el aire acondicionado, es promover un mundo sostenible, la creación de estándares y la educación continua.

Se ha utilizado el libro 2009 ASHRAE Handbook Fundamentals, que se compone de 39 capítulos divididos en diferentes temas como *Principles, Indoor Environmental Quality, Load and Energy Calculations, Materials...*

Los capítulos utilizados principalmente han sido:

- Chapter 17: **Residential cooling and heating load calculations**
- Chapter 18: **Nonresidential cooling and heating load calculations**

Con estos documentos se han calculado las pérdidas de calor en invierno y las ganancias de calor en verano, tanto para edificios residenciales como para edificios correspondientes a terciario. Estos dos capítulos se encuentran, íntegramente (y en su forma original-en inglés-), en el Anexo de este Trabajo Técnico.

Además se han tenido que consultar otros capítulos del citado libro como, *Climatic design information* (Ch. 14), *Fenestration* (Ch. 15) o *Ventilation and infiltration* (Ch. 16) entre otros.

2. ANTECEDENTES

La sociedad actual necesita un alto consumo energético para mantener su nivel de vida y confort. Como consecuencia de ello, existe una creciente preocupación por la conservación del medioambiente y en particular, por el cambio climático. El Protocolo de Kioto, es un instrumento internacional consensuado en 1997 y que establece objetivos legalmente obligatorios para que los países industrializados reduzcan las emisiones de los gases de efecto invernadero, contribuyendo así a la lucha contra el cambio climático. El fomento de la eficiencia energética en todos los sectores constituye una parte importante del conjunto de políticas y medidas necesarias para cumplir lo dispuesto en el Protocolo de Kioto.

Según los estudios realizados en la Unión Europea “el sector de la vivienda y de los servicios, compuesto en su mayoría por edificios, absorbe más del 40% del consumo final de energía en la Comunidad Europea y se encuentra en fase de expansión, tendencia que previsiblemente hará aumentar el consumo de energía y, por lo tanto, las emisiones de dióxido de carbono”.

Como consecuencia, surge la Directiva 2002/91/CE del Parlamento Europeo y del Consejo de la Unión Europea, de 16 de diciembre de 2002, relativa a la eficiencia energética de los edificios.

La transposición de la citada Directiva se ha realizado a través de tres pilares básicos que constituyen un punto de inflexión en el sector de la edificación: el Código Técnico de la Edificación, el Reglamento de Instalaciones Térmicas en los Edificios y el Real Decreto 47/2007, que establece la obligatoriedad de certificar energéticamente los edificios y proporcionar a los compradores y usuarios la calificación de la eficiencia energética obtenida.

El objetivo del nuevo marco normativo es reducir el consumo de energía en el sector residencial. Para ello, existen básicamente tres vías: el diseño ecoeficiente o bioclimático de las edificaciones, la incorporación de instalaciones de aprovechamiento de energías renovables y el fomento de la eficiencia y el ahorro energético a través de equipamientos eficientes, materiales innovadores y la concienciación de los usuarios. El resultado final debe ser por tanto, aumentar la eficiencia energética de los edificios, disminuyendo para ello el consumo de energía por unidad de servicio prestado.

2.1. NORMATIVA

La Directiva 2002/91/CE del Parlamento Europeo y del Consejo de la Unión Europea, de 16 de diciembre de 2002, relativa a la eficiencia energética de los edificios, tiene como objetivo fomentar la eficiencia energética de los edificios de la Comunidad Europea, teniendo en cuenta las condiciones climáticas exteriores y las particularidades locales, así como los requisitos ambientales interiores y la relación coste-eficacia.

La citada Directiva establece que los Estados miembros deben adoptar una metodología de cálculo de la eficiencia energética integrada de los edificios, aplicar requisitos mínimos de eficiencia energética en los edificios nuevos y en grandes edificios existentes que sean objeto de reformas importantes y poner a disposición de los compradores y usuarios un certificado de eficiencia energética, cuando los edificios sean construidos, vendidos o alquilados.

El concepto de eficiencia energética viene recogido en la Directiva 2002/91/CE como la cantidad de energía consumida realmente o que se estime necesaria para satisfacer las distintas necesidades asociadas a un uso estándar del edificio, que podrá incluir, entre otras cosas, la calefacción, el calentamiento del agua, la refrigeración, la ventilación y la iluminación. Dicha magnitud debe quedar reflejada en uno o más indicadores cuantitativos calculados teniendo en cuenta el aislamiento, las características técnicas y de la instalación, el diseño y la orientación, en relación con los aspectos climáticos, la exposición solar y la influencia de construcciones próximas, la generación de energía propia y otros factores, incluidas las condiciones ambientales interiores, que influyan en la demanda de energía.

La transposición de la Directiva Europea mencionada se ha realizado en gran medida, mediante el Documento Básico de Ahorro de Energía del Código Técnico de la Edificación, el Reglamento de Instalaciones Térmicas en los Edificios y el Real Decreto 47/2007, por el que se aprueba el Procedimiento básico para la certificación de eficiencia energética de edificios de nueva construcción.

El Código Técnico de la Edificación (CTE), establece las exigencias que deben cumplir los edificios en relación con los requisitos básicos de seguridad y habitabilidad, adoptando los códigos basados en prestaciones u objetivos.

El Código Técnico contiene un Documento Básico de Ahorro de Energía donde se establecen las exigencias básicas en eficiencia energética y energías renovables que deben cumplir los nuevos edificios y los que se reformen o rehabiliten. Este Documento Básico consta de las siguientes secciones:

- HE.1: Limitación de demanda energética.
- HE.2: Rendimiento de las instalaciones térmicas.
- HE.3: Eficiencia energética de las instalaciones de iluminación.

- HE.4: Contribución solar mínima de agua caliente sanitaria.
- HE.5: Contribución fotovoltaica mínima de energía eléctrica.

El Reglamento de Instalaciones Térmicas en los Edificios (RITE), establece las condiciones que deben cumplir las instalaciones destinadas a atender la demanda de bienestar térmico e higiene a través de las instalaciones de calefacción, climatización y agua caliente sanitaria, para conseguir un uso racional de la energía.

Las mayores exigencias en eficiencia energética que establece el RITE 2007, se concretan en:

- Mayor Rendimiento Energético en los equipos de generación de calor y frío, así como los destinados al movimiento y transporte de fluidos.
- Mejor aislamiento en los equipos y conducciones de los fluidos térmicos.
- Mejor regulación y control para mantener las condiciones de diseño previstas en los locales climatizados.
- Utilización de energías renovables disponibles, en especial la energía solar y la biomasa.
- Incorporación de subsistemas de recuperación de energía y el aprovechamiento de energías residuales.
- Sistemas obligatorios de contabilización de consumos en el caso de instalaciones colectivas.
- Desaparición gradual de combustibles sólidos más contaminantes.
- Desaparición gradual de equipos generadores menos eficientes.

La certificación energética de los edificios es una exigencia derivada de la Directiva 2002/91/CE. En lo referente a la certificación energética, esta Directiva se transpone parcialmente al ordenamiento jurídico español a través del Real Decreto 47/2007, de 19 de enero, por el que se aprueba el Procedimiento básico para la certificación de eficiencia energética de edificios de nueva construcción.

El objetivo del Procedimiento básico para la certificación de eficiencia energética de edificios es determinar la metodología de cálculo de la calificación de eficiencia energética, considerando aquellos factores que más incidencia tienen en el consumo de energía de los edificios, así como establecer las condiciones técnicas y administrativas para las certificaciones de eficiencia energética y aprobar un distintivo común en todo el territorio nacional denominado etiqueta de eficiencia energética.

En este certificado, y mediante una etiqueta de eficiencia energética, se asigna a cada edificio una clase energética de eficiencia, que varía desde la clase A, para los energéticamente más eficientes, a la clase G, para los menos eficientes. El certificado tiene una vigencia de 10 años.

3. OBJETIVOS

El diseño de edificios debe considerar los aspectos de ahorro de energía, por ejemplo poniendo ventanales amplios mirando al sur (en el hemisferio norte y en latitudes medias y altas) para que los días de invierno la radiación solar caliente los recintos, aplicando un aislante térmico a las superficies del edificio, especialmente aquellas que componen la envolvente térmica del edificio (cubiertas, fachadas, forjados, etc.), para disminuir las fugas de calor; o instalando paneles solares que aumenten la independencia de la energía eléctrica. La idea es construir edificios bioclimáticos encargados de aprovechar la energía del entorno.

3.1. ARQUITECTURA BIOCLIMÁTICA

3.1.1. DEFINICIÓN

La arquitectura bioclimática tiene como objetivo principal hacer un uso eficiente de la energía y de los recursos garantizando unas condiciones óptimas de confort y la sostenibilidad del medio ambiente.

Para ello, realiza un diseño inteligente que aprovecha las condiciones climáticas y los recursos disponibles del entorno y los transforma en confort interno, minimizando el impacto ambiental, como consecuencia de la reducción de la demanda y del consumo de energía.

A pesar de que parece un concepto nuevo, se lleva utilizando tradicionalmente desde hace siglos. Los pueblos indígenas siempre han observado los espacios naturales para ubicar las viviendas en lugares que permitiesen el máximo aprovechamiento de las condiciones climáticas del lugar. Ejemplo de ello es la arquitectura vernácula islámica, que utiliza el viento como generador de energía y el fenómeno físico de enfriamiento por evaporación, para acondicionar los espacios interiores. Otros ejemplos de arquitectura bioclimática son las casas encaladas en Andalucía o los tejados orientados al sur en el hemisferio Norte, con el objetivo de aprovechar la inclinación del Sol. En las casas rurales de muchas partes del mundo, el establo inferior servía de calefactor en invierno y de aislamiento térmico en verano, funcionando como una cámara de aire.

Mientras muchos pueblos del mundo siguieron viviendo en armonía con su entorno natural, en la cultura europea esta sabiduría se fue perdiendo paulatinamente sobre todo en las ciudades, a causa de la descoordinación y falta de regulación de las actuaciones públicas y privadas llegando a convertirse este olvido en un problema sanitario de primera magnitud.

El progreso científico-técnico del siglo XX marcado por las innovaciones tecnológicas, en el campo de la construcción, posibilitó el acceso a una vivienda salubre a una enorme cantidad de población en un siglo que vio el mayor incremento demográfico de la historia. Sin embargo, paralelamente, significó un elevado coste de desgaste en recursos naturales, en contaminación ambiental y en desastres naturales inducidos por el hombre, difícilmente asumibles a largo plazo.

Es precisamente en las sociedades desarrolladas cuando, a finales de siglo XX y una vez resueltos los problemas más acuciantes de asentamiento de la población y garantizado su nivel de vida, se asiste a un replanteamiento general de los procesos de producción industrial, y de sus consecuencias en costes energéticos y ambientales. La construcción, como uno de los factores claves del desarrollo industrial, es uno de los primeros objetos de estudio en el campo de la adecuación ambiental, produciéndose numerosos trabajos sobre el tema, a partir, sobre todo, de las crisis energéticas de los setenta.

Si en un principio los primeros trabajos realizados se dirigían exclusivamente al ahorro energético, en la actualidad el problema ha transcendido hacia el entendimiento de la arquitectura bioclimática como un proceso activo, donde la adecuación climática y ambiental se busca en todos los sentidos, desde los energéticos hasta los estéticos y funcionales.

Diferentes estudios avalan la capacidad de ahorro de energía de hasta un 70% en las soluciones arquitectónicas pensadas desde una lógica de adaptación y adecuación al clima, en comparación a los usuales procedimientos de construcción tradicional. En consecuencia, aplicando las normativas de ámbito nacional e internacional que evalúan el gasto energético en función de su equivalencia en gases contaminantes de efecto invernadero (CO₂), la utilización de la arquitectura bioclimática se presenta como una solución respetuosa con el medioambiente y efectiva para la lucha contra el cambio climático.

La utilización de la arquitectura bioclimática conduce también a un balance positivo en términos económicos a la hora de realizar una cuenta global: los índices de sobrecosto al aplicar lógicas de adecuación bioclimática a la construcción han implicado en España, una media de un 15% más sobre el coste de construcción de una vivienda tradicional, que son compensados sobradamente con los ahorros energéticos obtenidos, del orden de un 70%, y que, en el cómputo global relacionado con la vida útil del edificio, supone un ahorro neto total (costes de construcción más costes de mantenimiento) en torno al 20%.

Por tanto, la filosofía de esta arquitectura que apuesta por el uso inteligente de sistemas pasivos para disminuir las demandas energéticas y la integración de sistemas activos basados en fuentes de energía renovables, se traduce en ahorro energético, eficiencia energética y reducción del impacto ambiental.

3.1.2. TIPOS DE EDIFICACIÓN BIOCLIMÁTICA

Dependiendo de la extensión del balance energético global al que se refiere la adecuación climática y ambiental de la arquitectura, se pueden catalogar los distintos tipos de edificación bioclimática.

En el orden más sencillo, se encontrarían aquellos edificios que tienen como objetivo conseguir una alta eficiencia energética una vez construidos, sin considerar más variables ecológicas que relacionan el ambiente y la arquitectura, que las derivadas del ahorro energético a largo plazo.

En un segundo orden se situarían aquellos otros en donde el balance energético global incluye no sólo la fase de vida útil del edificio, sino todo su proceso constructivo, desde la extracción de los materiales, su elaboración industrial, su puesta en obra, su uso, su reciclaje y su destrucción.

En un tercer orden, se situarían aquellas edificaciones que no sólo se preocupan de mantener buenos balances energéticos, sino también en adecuarse al medio en un sentido más extenso, a través de la limitación del impacto visual o el mantenimiento de recursos naturales.

3.1.3. CONCEPTOS Y PRINCIPIOS BÁSICOS

La arquitectura bioclimática cuenta con dos tipos de sistemas, que habitualmente se utilizan combinados, para limitar el consumo energético. Se trata de los sistemas pasivos y los sistemas activos.

Los sistemas pasivos, son medidas que tienen como objetivo el control de las variables que afectan al confort térmico del interior de un edificio, basadas en un diseño adecuado a las condiciones del entorno. Estas medidas deben limitar la demanda energética, integrando para ello los recursos externos y garantizar un confort térmico, acústico y lumínico.

Los sistemas pasivos inciden especialmente, en el control de la radiación solar, facilitando o limitando su incidencia, en el aislamiento y en la inercia térmica.

Los sistemas activos, son aquellos que aplican las nuevas tecnologías de aprovechamiento de las energías renovables. Los más destacados son la energía solar (para cubrir la demanda de agua caliente sanitaria o de calefacción), la energía eólica, la energía geotérmica o la biomasa.

En este apartado también se incluyen sistemas que contribuyen al ahorro energético, como la domótica, los sistemas variables de iluminación, persianas, centrales de cogeneración, etc.

Algunos de los conceptos y principios básicos que se deben tener en cuenta en la arquitectura bioclimática son los siguientes:

Trayectoria solar y radiación

Siendo el sol la principal fuente energética que afecta al diseño bioclimático, es importante conocer su trayectoria en las distintas estaciones del año.

Las trayectorias solares tienen una consecuencia clara sobre la radiación recibida por fachadas verticales: en invierno, en el hemisferio Norte, la fachada sur recibe la mayoría de radiación, gracias a que el sol está bajo, mientras que las otras orientaciones apenas reciben radiación. En verano, en cambio, cuando el sol está más vertical a mediodía, la fachada sur recibe menos radiación directa, mientras que las mañanas y las tardes castigan especialmente a las fachadas este y oeste, respectivamente.

La energía solar incidente en una superficie terrestre se manifiesta de tres maneras diferentes:

- Radiación directa, la que proviene directamente del sol
- Radiación difusa, aquella recibida de la atmósfera como consecuencia de la dispersión de parte de la radiación del sol en la misma. Esta energía puede suponer aproximadamente un 15% de la radiación global en los días soleados, pero en los días nublados, supone un porcentaje mucho mayor.
- Radiación reflejada, aquella reflejada por la superficie terrestre. La cantidad de radiación depende del coeficiente de reflexión de la superficie, también llamado albedo.

Confort térmico

La sensación de confort térmico implica que el calor que genera el cuerpo se pueda transferir en un ritmo adecuado al medio exterior.

Los factores que influyen en el ritmo de generación de calor son los siguientes:

- Actividad física y mental: el cuerpo debe generar calor para mantener una determinada temperatura corporal, pero también influye el tipo de actividad física y mental que se realice.
- Metabolismo: cada persona tiene su propio metabolismo y por tanto, necesita su propio ritmo de evacuar calor. La tasa metabólica habitualmente se mide en met y en W/m^2 (1 met equivale a $58,15 W/m^2$).

Ubicación

El comportamiento climático de una casa no solo depende de su diseño, sino que también está influenciado por su ubicación: la existencia de accidentes naturales como

montes, ríos, pantanos, vegetación, o artificiales como edificios próximos, etc., crean un microclima que afecta al viento, la humedad y la radiación solar que recibe la casa.

Si se ha de construir una casa bioclimática, se deben estudiar las condiciones macroclimáticas de la región y las microclimáticas de la ubicación concreta. Las condiciones microclimáticas se pueden modificar con intención de conseguir un efecto beneficioso para el diseño bioclimático.

Las condiciones macroclimáticas son consecuencia de la pertenencia a una latitud y región determinada. Los datos más importantes que las definen son:

- Las temperaturas medias, máximas y mínimas
- La pluviometría
- La radiación solar incidente
- La dirección del viento dominante y su velocidad media

Las condiciones microclimáticas son consecuencia de la existencia de accidentes geográficos locales que pueden modificar las anteriores condiciones de forma significativa. Podemos tener en cuenta:

- La pendiente del terreno, la cual determina una orientación predominante de la vivienda.
- La existencia cercana de elevaciones, que pueden influir como barrera frente al viento o frente a la radiación solar.
- La existencia de masas de agua cercanas, que reducen las variaciones bruscas de temperatura e incrementan la humedad ambiente.
- La existencia de masas boscosas cercanas.
- La existencia de edificios, que impidan la incidencia de la radiación solar.

Forma y orientación

La forma del edificio influye sobre los siguientes aspectos:

- La superficie de contacto entre la vivienda y el exterior influye en las pérdidas o ganancias caloríficas. Normalmente se desea un buen aislamiento, para lo cual, además de utilizar los materiales adecuados, la superficie de contacto tiene que ser lo más pequeña posible. Para un determinado volumen interior, una forma compacta (como el cubo), sin entrantes ni salientes, es la que determina la superficie de contacto más pequeña. La existencia de patios, alas, etc. incrementan esta superficie. La medida que relaciona el volumen con la superficie de contacto con el medio exterior es la compacidad. En la arquitectura bioclimática interesa edificios con alta compacidad.
- La resistencia frente al viento que dependa de aspectos como la altura o la forma del tejado. Interesa llegar a un compromiso que disminuya las infiltraciones en invierno e incremente la ventilación en verano.
- La captación solar (se explica en el apartado de la orientación).

La orientación del edificio influye sobre los siguientes aspectos:

- La captación solar, en general interesa que sea lo más elevada posible porque es la fuente de climatización en invierno, mientras que en verano se utiliza elementos de sombreado que disminuyan la superficie captadora. La captación solar sucede principalmente a través de los acristalamientos, por tanto la mayor superficie acristalada se debe orientar hacia el sur.
- La influencia de los vientos dominantes sobre la ventilación y las infiltraciones.

Masa térmica

La masa térmica provoca un desfase entre los aportes de calor y el incremento de la temperatura. Funciona a distintos niveles: en ciclo diario, durante el invierno, la masa térmica estratégicamente colocada almacena el calor solar durante el día para liberarlo por la noche, y durante el verano, realiza la misma función, sólo que el calor que almacena durante el día es el de la casa (manteniéndola, por tanto, fresca), y lo libera por la noche, evacuándose mediante la ventilación. En ciclo interdiario, la masa térmica es capaz de mantener determinadas condiciones térmicas durante algunos días una vez que estas han cesado: por ejemplo, es capaz de guardar el calor de días soleados de invierno durante algunos días nublados venideros. En ciclo anual, se guarda el calor del verano para el invierno y el fresco del invierno para el verano (sólo una ingente masa térmica como el suelo es capaz de realizar algo así).

La vivienda con elevada masa térmica se comporta manteniendo una temperatura sin variaciones bruscas, relativamente estable frente a las condiciones externas. El objetivo es conseguir que, mediante un buen diseño bioclimático, esta temperatura sea agradable. La masa térmica elevada no es aconsejable en viviendas ocasionales (viviendas de fin de semana, por ejemplo), cuyas condiciones de temperatura son irrelevantes excepto en los momentos en que se ocupan, momentos en los que se requiere calentarlas o enfriarlas rápidamente.

En general, materiales de construcción pesados pueden actuar como una eficaz masa térmica: los muros, suelos o techos gruesos, de piedra, hormigón o ladrillo, son buenos en este sentido. Colocados estratégicamente para recibir la radiación solar tras un cristal, funcionan fundamentalmente en ciclo diario, pero repartidos adecuadamente por toda la casa, funcionan en ciclo interdiario. Si la casa está enterrada o semienterrada, la masa térmica del suelo ayudará también a la amortiguación de oscilaciones térmicas, en un ciclo largo.

Se puede realizar un aprovechamiento climático del suelo, gracias a su elevada inercia térmica que amortigua las oscilaciones térmicas del exterior. La temperatura del suelo suele ser tal que es menor que la temperatura exterior en verano, y mayor que la exterior en invierno, por lo que siempre tiene una influencia positiva. Por otro lado, la tierra actúa también como aislante.

Aislamiento y acristalamiento

El aislamiento térmico dificulta el paso de calor por conducción del interior al exterior de la vivienda y viceversa. Por ello es eficaz tanto en invierno como en verano. Una forma de conseguirlo es utilizar recubrimientos de materiales muy aislantes, como espumas y plásticos. Para hacer eficaz el aislamiento, también es necesario reducir al máximo los puentes térmicos.

La colocación del aislamiento es más efectiva si se hace por fuera de la masa de térmica, puesto que se aprovecha mejor la inercia térmica de los cerramientos.

También es importante aislar los acristalamientos. Al igual que en los cerramientos opacos, a través de los acristalamientos se produce una conducción de calor entre la zona fría y la zona caliente, lo que supone unas pérdidas energéticas importantes. En invierno, las pérdidas a través de las ventanas son notablemente superiores a las que se dan a través de los cerramientos aislados, por lo que se debe utilizar vidrios que sean lo más aislantes posibles. Un doble acristalado reduce las pérdidas de calor, al encerrar una cámara de aire, inmóvil y seco, que limita los intercambios térmicos por convección, aprovechando la baja conductividad térmica del aire. Sin embargo, reduce algo la transparencia frente a la radiación solar durante el día.

Las transferencias por radiación en los acristalamientos se pueden disminuir también utilizando vidrios revestidos con capas de baja emisividad, que bloquean la radiación al exterior a través del vidrio. Para disminuir la radiación solar que entra en verano debe combinarse con un vidrio exterior de control solar.

Otra opción eficaz es el uso de aislamientos móviles (contraventanas, persianas, paneles, cortinas) que se utilicen durante la noche y se quiten durante el día. Los elementos de sombreado fijos, como voladizos y retranqueos impiden la penetración de la radiación solar en verano, por lo que se debe tener especial cuidado en su diseño, de forma que se llegue a un compromiso entre el factor solar que se obtenga en invierno y en verano.

El factor solar de un acristalamiento es la relación entre la energía total que entra en un edificio a través del acristalamiento y la energía que incide en el mismo. Durante el invierno interesa que el factor solar sea lo mayor posible, contribuyendo a disminuir la demanda de calefacción. Por el contrario, en verano interesa limitar la radiación solar, a través de elementos de sombreado móviles, como cortinas o persianas, o fijos, como voladizos y retranqueos, que disminuyan el factor solar.

Se debe llegar a un compromiso entre las necesidades de demanda de calor y frío, considerando la trayectoria del sol y la orientación como factores clave a tener en cuenta en el diseño de las superficies acristaladas, sus propiedades térmicas y los elementos de sombreado.

También es muy importante tener en cuenta las carpinterías. En general se forma un puente térmico ya que la resistencia a la conducción del calor es menor en esta zona que en el cerramiento opaco y en el vidrio adyacente. Los marcos con rotura de puente térmico evitan las pérdidas de calor por este motivo y la formación de condensaciones en estas zonas, debido a que las temperaturas que hay en estas zonas son inferiores a las del interior la vivienda que se obtienen en estas zonas.

Captación solar pasiva

La energía solar es la fuente principal de energía de climatización en una vivienda bioclimática. Su captación se realiza aprovechando el propio diseño de la vivienda, y sin necesidad de utilizar sistemas mecánicos. La captación hace uso del llamado efecto invernadero, según el cual la radiación penetra a través de vidrio que es transparente a la radiación visible, calentando los materiales dispuestos detrás y no dejando escapar la radiación infrarroja emitida por estos materiales, por resultar opaco a la radiación de mayor longitud.

Los materiales, calentados por la energía solar, guardan este calor y lo liberan, posteriormente, atendiendo a un retardo que depende de su inercia térmica. Para un mayor rendimiento, es aconsejable disponer de sistemas de aislamiento móviles (persianas, contraventanas, etc.) que se puedan cerrar por la noche para evitar pérdidas de calor por conducción y convección a través del vidrio.

Los sistemas de captación pueden ser definidos por dos parámetros: rendimiento, o fracción de energía realmente aprovechada respecto a la que incide, y retardo, o tiempo que transcurre entre que la energía es almacenada y liberada. Hay varios tipos de sistemas:

- Sistemas directos, el sol penetra directamente a través del acristalamiento al interior del recinto. Es importante prever la existencia de masas térmicas de acumulación de calor en los lugares (suelo, paredes) donde incide la radiación. Son los sistemas de mayor rendimiento y de menor retardo.
- Sistemas semidirectos, utilizan un adosado o invernadero como espacio intermedio entre el exterior y el interior. La energía acumulada en este espacio intermedio se hace pasar a voluntad al interior a través de un cerramiento móvil. El rendimiento de este sistema es menor que el anterior, mientras que su retardo es mayor.
- Sistemas indirectos, la captación la realiza directamente un elemento de almacenamiento dispuesto inmediatamente detrás del cristal. El calor almacenado pasa al interior por conducción, convección y radiación. El elemento de almacenamiento puede ser un cerramiento de alta capacidad calorífica, bidones de agua, lecho de piedras, etc. Un caso particular es el llamado muro trombe, en el cual, una superficie captadora calienta el aire que hay en una cámara y este circula por convección a través de unas rejillas superiores e inferiores, que se abren y se cierran según las necesidades de calor o frío.

Ventilación e infiltraciones

En una vivienda bioclimática, la ventilación es importante, y tiene varios usos:

- Renovación del aire, para mantener las condiciones higiénicas. Un mínimo de ventilación es siempre necesario.
- Incrementar el confort térmico en verano, puesto que el movimiento del aire acelera la disipación de calor del cuerpo humano.
- Climatización, ya que el aire en movimiento puede llevarse el calor acumulado en cerramientos. Esto es útil especialmente en las noches de verano.

Infiltraciones, es el nombre que se le da a la ventilación no deseada. En invierno, pueden suponer una importante pérdida de calor, por ello es necesario reducirlas al mínimo controlando la estanqueidad de las carpinterías.

Existen diferentes formas de ventilar, actualmente y según el CTE, se exige ventilación híbrida o ventilación mecánica. La ventilación híbrida combina la ventilación natural y la mecánica.

- Ventilación natural, es la que tiene lugar cuando el viento crea corrientes de aire en la casa, al abrir las ventanas o cuando existe el efecto chimenea, por una diferencia de temperaturas entre el interior y el exterior.
- Ventilación convectiva o forzada, se basa en la diferencia de temperatura entre las masas de aire y tiene lugar cuando el aire caliente asciende, siendo reemplazado por el más frío. En las viviendas se pueden crear corrientes de aire mediante aperturas en los espacios superiores e inferiores o el uso de espacios tapón como el desván.

Aprovechamiento climático del suelo y los espacios tapón

La elevada inercia térmica del suelo provoca que las oscilaciones térmicas del exterior se amortigüen cada vez más según la profundidad. A una determinada profundidad, la temperatura permanece constante. La temperatura del suelo suele ser tal que es menor que la temperatura exterior en verano, y mayor que la exterior en invierno, con lo que siempre se agradece su influencia. Además de la inercia térmica, la tierra actúa como aislante.

Los espacios tapón son espacios adosados a la vivienda, de baja utilización, que térmicamente actúan de aislantes o "tapones" entre la vivienda y el exterior. El confort térmico en estos espacios no está asegurado, puesto que no forman parte de la vivienda propiamente dicha (el recubrimiento aislante no los incluirá). La colocación adecuada de estos espacios puede acarrear beneficios climáticos para la vivienda. Pueden ser espacios tapón el garaje, el invernadero, el desván...etc.

Pérdida de calor en viviendas (invierno)

En el interior de la casa, el calor se transmite entre los cerramientos (muros, techos, suelos) principalmente por radiación, y entre los cerramientos y el aire interior

principalmente por convección. El calor se transmite a través de los cerramientos por conducción, hasta alcanzar el exterior de la casa, donde se disipa por convección y radiación. Para reducir las pérdidas de calor, se actúa principalmente sobre el fenómeno de conducción a través de los cerramientos, intercalando una capa de material térmicamente aislante.

Para realizar el análisis energético de la envolvente térmica de un edificio, se trabaja habitualmente con flujos unidimensionales de calor a través de capas plano-paralelas. Sin embargo, existen ciertas zonas de la envolvente térmica que presentan discontinuidades, tanto geométricas (esquinas, aristas...) como materiales (interrupciones de la capa de aislante térmico). En estas zonas, llamadas puentes térmicos, el flujo de calor ya no presenta una distribución unidimensional, si no que se vuelve bi/tridimensional. Como resultado, aumenta la densidad del flujo térmico, concentrándose las pérdidas de calor en dichos puntos. Los puentes térmicos se pueden clasificar en función de si están integrados en fachada o no lo están: al primer caso pertenecen los puentes térmicos de contorno de hueco, caja de persiana y pilar; al segundo caso pertenecen los puentes térmicos de encuentro de fachada con forjado, con cubierta, con suelo exterior y con solera, así como esquinas entrantes y salientes

Sin embargo existe otra causa de pérdida de calor: la ventilación. Para que una casa sea salubre necesita un ritmo adecuado de renovación de aire. Si esta renovación se realiza con el aire exterior, se está perdiendo aire caliente e introduciendo aire frío. Hay que llegar a un compromiso entre la ventilación que se necesita y las pérdidas de calor que se pueden admitir.

Pero aunque se reduzca la ventilación al mínimo, una baja estanqueidad de la casa puede forzar la ventilación, especialmente en días ventosos: son las infiltraciones. Por ello, es importante reducir al máximo este fenómeno, cuidando especialmente las juntas de cierre de puertas y ventanas.

El viento, provoca convección forzada, haciendo que el calor se disipe mucho más rápidamente en los cerramientos en contacto con el exterior. Por ello, es aconsejable que la casa esté protegida de los vientos dominantes de invierno.

Protección contra la radiación de verano

Es evidente que en verano hay que reducir las ganancias caloríficas al mínimo. Ciertas técnicas utilizadas para el invierno (aislamiento, espacios tapón) contribuyen con igual eficacia para el verano. Otras técnicas, como la ventilación, ayudan casi exclusivamente en verano. Sin embargo, los sistemas de captación solar pasiva, tan útiles en invierno, son ahora perjudiciales, por cuanto es necesario impedir la penetración de la radiación solar, en vez de captarla.

Afortunadamente, en verano el sol está más alto que en invierno, lo cual dificulta su penetración en las cristaleras orientadas al sur. La utilización de un voladizo o retranqueo dificulta la penetración de la radiación directa, afectando poco a la penetración invernal. También el propio comportamiento del vidrio otorga beneficios, porque con ángulos de incidencia de la radiación más oblicuos, el coeficiente de transmisión es menor. A pesar de estos beneficios, existen tres inconvenientes:

- El solsticio de verano (21 de junio) no coincide exactamente con los días más calurosos del verano, lo que implica que el sol está algo más bajo cuando hace más calor.
- El día tiene mayor duración y son más despejados que en invierno
- Aunque se evite la radiación directa, hay que considerar también el efecto de la radiación difusa y reflejada.

Para impedir la radiación directa del sol se pueden emplear algunos de estos dispositivos de sombreado:

- Voladizo y retranqueos, con unas dimensiones adecuadas que impida algo de incidencia solar en verano y no estorbe mucho en invierno.
- Toldos, lamas y otros dispositivos externos, que presentan la ventaja de poder ajustarse a las condiciones requeridas.
- Alero enrejado con vegetación de hoja caduca, el ciclo vital de las plantas de hoja caduca coincide con el verano.
- Persianas exteriores.
- Contraventanas, aunque afectan a la entrada de luz.
- Árboles de hoja caduca, en verano ofrecen sombra y refrescan el ambiente por evapotranspiración.

Además de las técnicas anteriores, se pueden utilizar colores para las fachadas poco absorbentes de la luz solar, espacios tapón (espacios no habitables como cámaras de aire, desván, garaje) y sistemas evaporativos de refrigeración. El efecto deseado conseguir que la evaporación del agua refresque el ambiente.

Instalaciones e integración de energías renovables

Aunque el diseño del edificio se haya realizado mediante criterios bioclimáticos, minimizando la demanda de energía, es necesario cubrirla de la forma más eficiente posible y con el menor impacto para el medio ambiente. Para ello se deben utilizar equipos y sistemas de alta eficiencia energética y con altos rendimientos. Algunos sistemas utilizados son:

- Suelo radiante, que al operar con una temperatura del agua más moderada, disminuye las pérdidas por distribución y garantiza un mayor confort ya que no se producen corrientes de aire, manteniendo además, una temperatura homogénea en todo el espacio climatizado.

- Máquina de absorción para refrigeración, es un sistema eficiente que tiene un menor consumo eléctrico ya que no utiliza un compresor.
- Calderas de baja temperatura, pueden adaptar su temperatura de trabajo a las necesidades reales de la instalación en función de la demanda, por lo que son sistemas muy eficientes.
- Calderas de condensación, aprovechan la energía latente del vapor de agua para convertirlo en calor sensible y reducen las pérdidas de energía al disminuir la temperatura de los gases producto de la combustión.
- La cogeneración, es un procedimiento de alta eficiencia energética que aprovecha la energía térmica que se desprende al generar electricidad a partir de un combustible. En la trigeneración, se obtiene además calor sustraído del proceso de refrigeración del edificio.
- Recuperadores de calor, consiguen que el aire de expulsión del interior transfiera su calor al aire de impulsión en invierno y al contrario en verano, favoreciendo la disminución de la demanda tanto en el periodo de calefacción como de refrigeración.

Por otro lado, las nuevas tecnologías de aprovechamiento de las energías renovables, fomentan el ahorro energético y la disminución de emisiones de gases contaminantes. Las energías más utilizadas en el sector de la edificación son las siguientes:

- La energía solar, se obtiene directamente del Sol y se puede aprovechar mediante los sistemas pasivos de captación solar o los sistemas activos, entre los que se encuentra la energía solar térmica y la fotovoltaica. Principalmente se usa para producir agua caliente sanitaria, pero también puede contribuir a la demanda de calefacción o incluso refrigeración, mediante una máquina de absorción.
- Biomasa, para cubrir la demanda energética de un edificio se pueden utilizar calderas de biomasa, cuyo combustible tiene origen biológico.
- Bomba de calor geotérmica, permite reducir el consumo energético de un edificio, utilizando la inercia térmica del suelo, para absorber calor en invierno y cederlo en verano.

Es posible que todo el consumo de energía sea de generación propia y no contamine. En este caso, hablamos de “edificios 0 emisiones”. Incluso se puede llegar a generar más energía de la consumida, en cuyo caso hablamos de “edificios energía plus”.

4. RECOPIACIÓN Y PRESENTACIÓN DE DATOS

Desde el primer momento en que se supo el tipo de plugin a “enchufar” al programa, hubo que hacer un recuento de todos aquellos parámetros necesarios para su correcta implantación. Después de estudiar minuciosamente los capítulos 17 y 18 del libro Fundamentals 2009 de ASHRAE, se sacaron dos grandes conclusiones: la primera fue que el número de parámetros era excesivamente elevado y la segunda, que la construcción y arquitectura estadounidense y española diferían bastante, por tanto, muchas de las tablas que la sociedad americana utilizaba no se correspondían exactamente con la realidad española.

En los siguientes apartados se mostrarán los datos que son necesarios para poder realizar el plugin correctamente, sin desviarse del método y además, las dificultades que aparecieron a lo largo de todo el proceso.

4.1. DATOS E INFORMACIÓN NECESARIA PARA LA REALIZACIÓN DEL PLUGIN

Un objetivo propuesto al principio del diseño fue la utilización, al máximo, de los datos que el usuario introduce conforme está realizando la certificación del edificio, con el fin de no sobrecargar demasiado la ventana que aparece cuando se hace clic sobre el plugin ASHRAE. De esta forma, el usuario va rellenando normalmente las distintas casillas de todos los paneles del programa y, sin darse cuenta, está completando también el proceso que le llevará a obtener el dimensionamiento de las instalaciones de calefacción y refrigeración de manera casi inmediata. Esto es así porque no se quiso que la ventana que apareciese pidiera excesiva información, ya que podía ocurrir que la certificación pareciese complicada y difícil.

Y como éstas son unas características que CE³X quiere evitar, los datos que pide el plugin son muy sencillos y fundamentales para completar los cálculos.

De **CE3X-Residencial**, se obtienen diferentes datos. De la envolvente, los siguientes:

- Todos y cada uno de los cerramientos que el usuario ha introducido en el programa. De éstos se obtiene el nombre, de qué tipo son, su área, su transmitancia, su orientación y con qué están en contacto.
- De la misma forma, todos los Puentes Térmicos, y de ellos su nombre, su conductancia (fi) y su longitud.
- Los huecos que existen en la envolvente. De éstos su nombre, de qué tipo son, la superficie, la transmitancia, el factor solar, la transmitancia del marco y su porcentaje respecto del hueco total, la orientación, si tiene o no protección solar, su altura, la absorptividad y el cerramiento en el que se sitúan.

Además, datos geográficos como:

- La provincia y la ciudad en que se sitúa el edificio.
- Su latitud.
- Las temperaturas de diseño del edificio (en función de dónde se ubique).

También son necesarios datos del propio edificio:

- El año de construcción del edificio.
- El tipo de edificio, ya sea bloque, unifamiliar o vivienda individual.
- La superficie útil.
- La altura de planta.
- La masa de las particiones, ya sea ligera, media o pesada.

Por último, los datos que se le piden al usuario en la ventana del plugin son:

- La ocupación que existe en dicha vivienda (número de personas).
- El número total de baños.
- El número total de cocinas.

De **CE3X-Terciario (ya sea Pequeño o Grande)**, se obtienen los siguientes datos. De la envolvente, los siguientes:

- Todos y cada uno de los cerramientos que el usuario ha introducido en el programa. De éstos se obtiene el nombre, de qué tipo son, su área, su transmitancia, su peso por unidad de área, su orientación y con qué están en contacto.
- De la misma forma, todos los Puentes Térmicos, y de ellos su nombre, su conductancia (fi) y su longitud.
- Los huecos que existen en la envolvente. De éstos su nombre, de qué tipo son, la superficie, la transmitancia, el factor solar, la transmitancia del marco y su porcentaje respecto del hueco total, la orientación, si tiene o no protección solar, su altura, la absorptividad y el cerramiento en el que se sitúan.

Además, datos geográficos como:

- La provincia y la ciudad en que se sitúa el edificio.
- Su latitud y su longitud.
- Las temperaturas de diseño del edificio (en función de dónde se ubique).

También son necesarios datos del propio edificio:

- El año de construcción del edificio.
- El perfil de uso del edificio.
- La superficie útil.
- La altura de planta.
- La masa de las particiones, ya sea ligera, media o pesada.

Por último, los datos que se le piden al usuario en la ventana del plugin son:

- La ocupación que existe en dicho edificio (número de personas).
- El horario que tiene el edificio.
- La potencia de los equipos (W).

4.2. DIFICULTADES EN LA IMPLANTACIÓN DEL MÉTODO

Si bien una de las cosas buenas que tienen los documentos de la ASHRAE es su facilidad para entenderlos, una vez que los has estudiado y analizado, no es menos cierto que la implantación de sus métodos no haya resultado inmediata, sino que ha habido que estudiar cómo poder utilizar sus múltiples tablas y factores para que sean eficaces. Se va a diferenciar entre residencial y terciario para explicar qué partes han sido las más difíciles de implementar.

Dificultades en residencial

Para Residencial, se ha empleado el **Residential Load Factor (RLF) Method**, que debe ser aplicado en el rango siguiente:

- Latitud entre 20° y 60°.
- Altura menor a 2000m.
- Clima cálido.

Estos parámetros se cumplen en su totalidad, pero existen otros que son más específicos como el tipo de construcción que no siempre se podrán cumplir.

Para seguir implementando el método es necesario fijar unas condiciones de diseño interiores y exteriores. No existen problemas con las condiciones interiores, propuestas por ASHRAE, y fijadas en 24°C y una humedad entre 50% y 60% para verano, y 20°C y humedad aproximada del 30% para temporada de calefacción. Para las condiciones exteriores, ASHRAE fija, en refrigeración, una temperatura de diseño de bulbo húmedo del 1% y coincidente con bulbo seco. Para calefacción, utiliza una temperatura de diseño de bulbo seco del 99%. Aquí se sigue el método exactamente.

Para la realización del proyecto se han utilizado Niveles de Percentiles Anuales (NPA), tanto de invierno como de verano, por considerarse que los datos de percentiles anuales son los más representativos y los más utilizados a nivel mundial, según la *Guía Técnica: condiciones climáticas exteriores de proyecto* (IDAE).

El cálculo de la potencia de calefacción se realiza de la manera habitual a la que se puede encontrar en cualquier manual de calefacción.

En refrigeración, para calcular la potencia de calor que entra a través de las superficies opacas, aparecen unos factores, opaque-surface cooling factors (Tabla 7) que dependen del tipo de superficie. Aquí ya aparecen tipos de construcciones típicamente americanas, como muros con estructura de madera, que serían impensables en la inmensa mayoría de las provincias españolas. Para el cálculo del calor a través de los huecos, existe un factor, Interior Shading Attenuation coefficient (IAC), que se calcula a partir del tipo de vidrio y con diferentes tipos de dispositivos como cortinas, stores o persianas. Ya que no se quería sobrecargar el número de datos que se le iban a pedir al usuario, se ha

optado por mantener ese dato igual a 1, para que no interfiera en el resultado. La dificultad se añadía porque vinculaba todos los dispositivos de sombreado anterior con los vidrios más típicos utilizados en Norteamérica, que no siempre son los utilizados aquí.

Dificultades en terciario

En cuanto al cálculo en edificios del sector terciario, ocurre algo similar que en residencial, y es que en los cálculos de calefacción, el método es el habitual pero en refrigeración, los cálculos son muy exclusivos. Para refrigeración se utiliza el **Radiant Time Series (RTS) Method**.

Uno de los problemas iniciales fue encontrar qué parte radiante y qué parte convectiva se asigna a los diferentes tipos de luminarias que existen en CE³X. No hubo problema con las halógenas y fluorescentes, pero sí con Sodio blanco, Vapor de mercurio, Halogenuros metálicos, Inducción y LED.

En este método son necesarias también las temperaturas, hora a hora, de las distintas ciudades en los meses de verano. Esto sí que se pudo conseguir.

Para el cálculo horario de los diferentes ángulos y sus posteriores irradiancias, es necesario tener 2 factores, τ_{ab} y τ_{ad} , que son los clear sky optical depth for beam or diffuse irradiance. En el capítulo 14 podemos obtener estos factores para el ejemplo del capítulo 18 (en Atlanta), pero se ha hecho imposible obtenerlo para cualquier otra ciudad, por lo que se ha optado por dejarlo con el mismo valor para todas las ciudades e igual a los valores para Atlanta.

Pero lo más difícil de implementar en este método, y que a la vez resulta fundamental para obtener resultados fiables son 2 aspectos:

- Los Conduction Time Factors (CTF) de muros y tejados:
El método ofrece hasta 35 diferentes construcciones de muros y otras 19 diferentes de tejados, con sus respectivos CTF hora a hora, muy diferentes entre ellos, pero que son fundamentales para saber cuando se produce la hora de máxima ganancia de calor.
El verdadero problema es que dichas construcciones son tan específicas que se han tenido que agrupar según su transmitancia y su peso por unidad de área, por lo que su fiabilidad puede verse reducida, aunque su transcendencia en el resultado final, será siempre pequeña.
- Los Radiant Time Factors (RTF) para las distintas construcciones:
En este caso ocurre algo muy parecido a lo anterior. Ahora existen solo dos tipos, Nonsolar RTF o Solar RTF (su diferencia es muy clara y se explica muy bien), pero dentro de ellos existen varias tablas en función del tipo de construcción, si tienen ó no alfombra interior y del porcentaje de huecos del

cerramiento. Finalmente se asume que no existe alfombra interior y que el porcentaje de huecos es del 50%.

5. DESARROLLO DEL TRABAJO

En este apartado se va a exponer completamente el código fuente de los archivos que se han programado para la realización de este trabajo técnico. Los ficheros principales son: ASHRAE.py, ASHRAETerciario.py y TablasASHRAE.

PluginsDES.py, que es el fichero donde se ubican todos los demás plugins del programa, es más extenso que lo que se ofrece a continuación, ya que solo aparece lo que interesa.

5.1. CÓDIGO FUENTE DE LOS FICHEROS PROGRAMADOS

5.1.1. PluginsDES.py:

```
# -*- coding: cp1252 -*-
import wx
import datosEdificio
import math
import sys
import os
Directorio = os.getcwd ()

import betacomb.BetacombDialogo as BetacombDialogo
import cargaEquiposRefrigeracion.RepartoCargaDialogo as RepartoCargaDialogo
import traduceCalenerVyp.importaCalenervypCEX as importaCalenervypCEX

import betacomb.listaZonas as listaZonas
import betacomb.ASHRAE as ASHRAE
import betacomb.ASHRAETerciario as ASHRAETerciario

class ciudad:
    def __init__(self,zona):
        if zona == 'A3':
            self.tVerano = 32
            self.tInvierno = 2
            self.gd = 227
        elif zona == 'A4':
            self.tVerano = 30
            self.tInvierno = 5
            self.gd = 208
        elif zona == 'B3':
            self.tVerano = 32
            self.tInvierno = 0
            self.gd = 516
        elif zona == 'B4':
            self.tVerano = 40
            self.tInvierno = 1
            self.gd = 438
        elif zona == 'C1':
            self.tVerano = 30
            self.tInvierno = 0
            self.gd = 820
        elif zona == 'C2':
            self.tVerano = 31
            self.tInvierno = 2
            self.gd = 656
        elif zona == 'C3':
            self.tVerano = 36
            self.tInvierno = -2
            self.gd = 1042
        elif zona == 'C4':
            self.tVerano = 34
            self.tInvierno = -4
            self.gd = 158
        elif zona == 'D1':
            self.tVerano = 26
            self.tInvierno = -4
```

```
        self.gd = 1560
    elif zona == 'D2':
        self.tVerano = 32
        self.tInvierno = -6
        self.gd = 1501
    elif zona == 'D3':
        self.tVerano = 34
        self.tInvierno = -3
        self.gd = 1405
    elif zona == 'E1':
        self.tVerano = 30
        self.tInvierno = -6
        self.gd = 2048
    else:
        self.tVerano = ""
        self.tInvierno = ""
        self.gd = ""

def create (parent):
    return menuItem(parent)
[wxID_P1_ITEM1, wxID_P1_ITEM2, wxID_P1_ITEM3, wxID_P1_ITEM4, wxID_P1_ITEM5,
 wxID_P1_ITEM6, wxID_P1_ITEM7, wxID_P1_ITEM8, wxID_P1_ITEM9, wxID_P1_ITEM10,
 wxID_P1_ITEM11, wxID_P1_ITEM12, wxID_P1_ITEM13, wxID_P1_ITEM14, wxID_P1_ITEM15,
 wxID_P1_ITEM16, wxID_P1_ITEM17, wxID_P1_ITEM18, wxID_P1_ITEM19, wxID_P1_ITEM20,
 wxID_P1_ITEM21, wxID_P1_ITEM22, wxID_P1_ITEM23, wxID_P1_ITEM24, wxID_P1_ITEM25,
 wxID_P1_ITEM26, wxID_P1_ITEM27, wxID_P1_ITEM28, wxID_P1_ITEM29, wxID_P1_ITEM30
] = [wx.NewId() for __init__ in range(30)]

class menuItem(wx.Menu):
    def __init__(self,parent):
        self.parent=parent
        self.menutit=wx.Menu ()

        self.menutit.Append (help="", id=wxID_P1_ITEM1, kind=wx.ITEM_NORMAL,
            text=u'Borrar Puentes Térmicos')

        self.menutit.Append (help="", id=wxID_P1_ITEM8, kind=wx.ITEM_NORMAL,
            text=u'Borrar Instalaciones')

        self.menutit.Append (help="", id=wxID_P1_ITEM2, kind=wx.ITEM_NORMAL,
            text=u'Parámetros adimensionales')

        self.menutit.Append (help="", id=wxID_P1_ITEM3, kind=wx.ITEM_NORMAL,
            text=u'Pon mis datos administrativos')

        self.menutit.Append (help="", id=wxID_P1_ITEM4, kind=wx.ITEM_NORMAL,
            text=u'Rellena precios de la energía (por defecto)')

        self.menutit.Append (help="", id=wxID_P1_ITEM5, kind=wx.ITEM_NORMAL,
            text=u'Estimación Potencia Calor/Frío')

        self.menutit.Append (help="", id=wxID_P1_ITEM6, kind=wx.ITEM_NORMAL,
            text=u'Estimación Fracción potencia por caldera y betacomb')
```

```
self.menuit.Append (help="", id=wxID_P1_ITEM7, kind=wx.ITEM_NORMAL,
                    text=u'Estimación Reparto Carga Refrigeración')

self.menuit.Append (help="", id=wxID_P1_ITEM24, kind=wx.ITEM_NORMAL,
                    text=u'Importa geometria Calener Vyp')

self.menuit.Append (help="", id=wxID_P1_ITEM25, kind=wx.ITEM_NORMAL,
                    text=u'Listado de Zonas')

self.menuit.Append (help="", id=wxID_P1_ITEM26, kind=wx.ITEM_NORMAL,
                    text=u'Demandas')

self.menuit.Append (help="", id=wxID_P1_ITEM27, kind=wx.ITEM_NORMAL,
                    text=u'Demandas Medidas de Mejora')

self.menuit.Append (help="", id=wxID_P1_ITEM28, kind=wx.ITEM_NORMAL,
                    text=u'Porcentajes de sombreamiento')

self.menuit.Append (help="", id=wxID_P1_ITEM29, kind=wx.ITEM_NORMAL,
                    text=u'Parámetros adimensionales Terciario')

self.menuit.Append (help="", id=wxID_P1_ITEM30, kind=wx.ITEM_NORMAL,
                    text=u'ASHRAE')

parent.Bind (wx.EVT_MENU, self.onMenuClick, id=wxID_P1_ITEM1)
parent.Bind (wx.EVT_MENU, self.onMenuClick2, id=wxID_P1_ITEM2)
parent.Bind (wx.EVT_MENU, self.onMenuClick3, id=wxID_P1_ITEM3)
parent.Bind (wx.EVT_MENU, self.onMenuClick4, id=wxID_P1_ITEM4)
parent.Bind (wx.EVT_MENU, self.onMenuClickEstimacionCalefaccion, id=wxID_P1_ITEM5)
parent.Bind (wx.EVT_MENU, self.onMenuClickEstimacionBetacomb, id=wxID_P1_ITEM6)
parent.Bind (wx.EVT_MENU, self.onMenuClickBorrarInstalaciones, id=wxID_P1_ITEM8)
parent.Bind (wx.EVT_MENU, self.onMenuClickEstimacionRepartoCargaRefrigeracion,
id=wxID_P1_ITEM7)
parent.Bind (wx.EVT_MENU, self.onMenuClickImportaGeometriaCalenerVyp,
id=wxID_P1_ITEM24)
parent.Bind (wx.EVT_MENU, self.onMenuClickListadoZonas, id=wxID_P1_ITEM25)
parent.Bind (wx.EVT_MENU, self.onMenuClickDemandas, id=wxID_P1_ITEM26)
parent.Bind (wx.EVT_MENU, self.onMenuClickDemandasMedidasMejora,
id=wxID_P1_ITEM27)
parent.Bind (wx.EVT_MENU, self.onMenuClickPorcentajesSombra, id=wxID_P1_ITEM28)
parent.Bind (wx.EVT_MENU, self.onParametrosAdimensionalesTerciario,
id=wxID_P1_ITEM29)
parent.Bind (wx.EVT_MENU, self.onCalculosASHRAE, id=wxID_P1_ITEM30)

def onCalculosASHRAE (self, event):
    if self.parent.programa == 'Residencial':
        dlg = ASHRAE.Dialog1(self.parent)
        dlg.ShowModal ()
    else:
        dlg = ASHRAETerciario.Dialog1(self.parent)
        dlg.ShowModal ()
```

5.1.2. ASHRAE.py:

```
# -*- coding: cp1252 -*-
#Boa: Dialog: Dialog1

import wx
from TablasASHRAE import *
from math import *

    ##si importo así, tengo todas las funciones de
    ##TablasASHRAE disponibles.Solo con poner su nombre me basta

    ##Si pusiera import TablasASHRAE
    ##luego tendría que poner: TablasASHRAE.nombreFuncion

def create (parent):
    return Dialog1 (parent)

[wxID_DIALOG1, wxID_DIALOG1CALCULARBUTTON,
wxID_DIALOG1CALCULARTERCIARIOBUTTON,wxID_DIALOG1OCUPACIONTEXT,wxID_PA
NEL1ACCESORIOTEXT,wxID_PANEL1ACCESORIOCHOICE, wxID_PANEL1TITULOTXT,
wxID_BANOSTEXT, wxID_DIALOG1BANOS, wxID_COCINASTEXT, wxID_DIALOG1COCINAS,
wxID_DIALOG1DEINICIOTEXT, wxID_DIALOG1HORARIOINICIOCHOICE,
wxID_DIALOG1DEFINTEXT, wxID_DIALOG1HORARIOFINCHOICE,
wxID_DIALOG1HORASFINTEXT, wxID_DIALOG1PERSONASTEXT,
wxID_DIALOG1OCUPACION, wxID_DIALOG1HORASINICIOTEXT
] = [wx.NewId () for _init_ctrls in range (19)]

class Dialog1 (wx.Dialog):

    def _init_ctrls(self, prnt):

        wx.Dialog.__init__(self, id=wxID_DIALOG1, name="", parent=prnt,
            pos=wx.Point(270, 100), size=wx.Size(501, 350),
            style=wx.DEFAULT_DIALOG_STYLE, title='Residential Load Calculations')
        self.SetClientSize (wx.Size (485, 300))
        self.SetBackgroundColour ('white')

        self.tituloTxt = wx.StaticText (id=wxID_PANEL1TITULOTXT,
            label=u'Cálculo de la potencia de Calefacción/Refrigeración', name=u'tituloTxt',
            parent=self, pos=wx.Point(25, 24), size=wx.Size(153, 16),
            style=0)
        self.tituloTxt.SetFont (wx.Font (12, wx.SWISS, wx.ITALIC, wx.BOLD,
            False, u'Georgia'))
        self.tituloTxt.SetForegroundColour (wx.Colour (0, 64, 128))

        self.calcularButton = wx.Button (id=wxID_DIALOG1CALCULARBUTTON,
            label=u'Calcular', name=u'calcularButton', parent=self,
            pos=wx.Point(188, 230), size=wx.Size(100, 32), style=0)
        self.calcularButton.Bind (wx.EVT_BUTTON, self.onCalcular,
            id=wxID_DIALOG1CALCULARBUTTON)

        self.banosText = wx.StaticText (id=wxID_BANOSTEXT,
            label=u'Nºxfamero total de baños', name=u'banosText', parent=self,
            pos=wx.Point(40, 166), size=wx.Size(130, 13), style=0)

        self.banos = wx.TextCtrl (id=wxID_DIALOG1BANOS,
            name=u'banos', parent=self, pos=wx.Point(176, 162),
```

```
size=wx.Size(40, 20), style=0, value=u")

self.cocinasText = wx.StaticText (id=wxID_COCINASTEXT,
    label=u'\xfamero total de cocinas', name=u'cocinasText', parent=self,
    pos=wx.Point(40, 123), size=wx.Size(130, 13), style=0)

self.cocinas = wx.TextCtrl (id=wxID_DIALOG1COCINAS,
    name=u'cocinas', parent=self, pos=wx.Point(176, 119),
    size=wx.Size(40, 20), style=0, value=u")

self.ocupacionText = wx.StaticText (id=wxID_DIALOG1OCUPACIONTEXT,
    label=u'Ocupaci\xf3n', name=u'ocupacionText', parent=self,
    pos=wx.Point(40, 80), size=wx.Size(50, 13), style=0)

self.ocupacion = wx.TextCtrl (id=wxID_DIALOG1OCUPACION,
    name=u'ocupacion', parent=self, pos=wx.Point(176, 76),
    size=wx.Size(40, 20), style=0, value=u")

self.personasText = wx.StaticText (id=wxID_DIALOG1PERSONASTEXT,
    label=u'personas', name=u'personasText', parent=self,
    pos=wx.Point(230, 80), size=wx.Size(50, 13), style=0)

def __init__(self, parent):
    self.parent = parent
    self._init_ctrls (parent)

#####

# CÁLCULO DE CARGAS DE CALEFACCIÓN Y REFRIGERACIÓN
# Según ASHRAE: 2009 Handbook - Fundamentals
# Todas las tablas utilizadas se encuentran en: Plugins\betacomb\TablasASHRAE.py

### Ref = Refrigeración
### Cal = Calefacción

##### Residential cooling and heating load calculations (Chapter 17) #####

#Cuando se pulsa el botón Calcular, llama a esta función:
def onCalcular (self, event):

#que a su vez llama a las siguientes, las cuales recogen todos los datos que hacen falta para realizar los
#calculos:

a = self.cogerCerramientos ()

b=self.cogerVentanas ()
```



```
#En datos se encuentran datos importantes para la realizacion de los calculos
c=self.datos ()
```

```
d = self.cogerPT ()
```

```
#aqui se encuentran los datos introducidos en la ventana por el usuario
e = self.cogerDatos ()
```

```
##### Carga sensible
```

```
## 1.-Cálculo de la potencia de Refrigeración/Calefacción a través de la ENVOLVENTE:
```

```
#1.1.POTENCIA PUENTES TÉRMICOS (sólo calefacción)
```

```
potPT=0
```

```
#voy recorriendo los PT
```

```
for i in range (len(self.parent.panelEnvolvente.puentesTermicos)):
```

```
    #Para cada PT:
```

```
    #->PT = fi * L * dT (W)
```

```
    #fi=conductividad lineal
```

```
    #L=longitud del PT
```

```
    #dT=salto termico calefaccion= 20 - Temp Diseño Calefaccion
```

```
    PT = float (d[i] [1]) * float (d[i] [2]) * (20 - float(c [3]))
```

```
    #->suma de cada PT
```

```
    potPT += PT
```

```
#1.2.POTENCIA SUPERFICIES OPACAS
```

```
potSupOpacasRef=0
```

```
potSupOpacasCal=0
```

```
#voy recorriendo los cerramientos:
```

```
for i in range (len(self.parent.panelEnvolvente.cerramientos)):
```

```
    #Para cada cerramiento:
```

```
    #->hallo los coeficientes de Refrigeracion: [OFt, OFb, OFr]
```

```
    #ASHRAE: Tabla 7 (pag 17.9)
```

```
    coeficientes = tabla7(a[i][1],a[i][-1],0.65,a[i][4])
```

```
    ## (cerramiento, con que está en contacto, AbsortanciaTecho, Orientacion)
```

```
    #->hallo el factor CFopq
```

```
    #CFopq = OFt*dT + OFb + OFr*DR
```

```
    #ASHRAE: Ecuacion 21 (pag 17.8)
```

```
    #dT = salto termico refrigeracion = Temperatura diseño refrigeracion - 24
```

```
    #DR = Daily Range = Oscilacion Media Diaria (OMD) = c [4]
```

```
    CFopq = coeficientes [0]*(float(c [2])-24) + coeficientes [1] + (coeficientes [2]*c [4])
```

```
    #potRef = A * U * CFopq (W)
```

```
    xRef = float(a[i][2]) * float(a[i][3]) * CFopq
```

```
    #potSupOpacasRef = suma de la potRef de cada cerramiento
```

```
    potSupOpacasRef += xRef
```

```
    #potCal = A * U * dT
```

```
    xCal = float(a[i][2]) * float(a[i][3]) * (20 - float(c[3]))
```

```
#potSupOpacasCal = suma de la potCal de cada cerramiento  
potSupOpacasCal += xCal
```

#1.3.POTENCIA HUECOS (puertas y ventanas)

```
potPuertasRef=0  
potPuertasCal=0  
potHuecosTotalesRef = 0  
potHuecosTotalesCal = 0  
#voy recorriendo los huecos:  
for i in range(len(self.parent.panelEnvolvente.ventanas)):  
    #si porcentajeMarco=100 ---> es puerta  
    if b[i] [7]=='100':  
        #Para cada puerta:  
        #->hallo los coeficientes Refrigeracion: [OFt, OFb, OFr]  
        #ASHRAE: Tabla 7 (pag 17.9)  
        coeficientesPuertas = tabla7a(b[i][1],b[i][7],b[i][6],b[i][8])  
        ##(cerramiento,%Marco,Orientacion,proteccionSolar)  
  
        #->hallo el factor CFopq (igual que antes)  
        #ASHRAE: Ecuacion 21 (pag 17.8)  
        CFopqPuerta = coeficientesPuertas [0]*(float(c [2]) - 24) + coeficientesPuertas [1] +  
        (coeficientesPuertas [2]*c [4])  
  
        #potPuertaRef = A * U * CFopqPuerta  
        yRef = float(b[i][5]) * float(b[i][2]) * CFopqPuerta  
  
        #potPuertasRef = suma de la potPuertaRef de cada puerta  
        potPuertasRef += yRef  
  
        #potPuertaCal = A * U * dT  
        yCal = float (b[i] [5]) * float (b[i] [2]) * (20 - float (c [3]))  
  
        #potPuertasCal = suma de la potPuertaCal de cada puerta  
        potPuertasCal += yCal  
  
    #si es ventana,  
    else:  
        #Para cada ventana se debe hacer todo lo siguiente:  
        #->si la orientacion es Techo----->Lucernario  
        if b[i][6]=='Techo':  
            #SLF = Shade Line Factors  
            #es un factor para hallar Fshd  
            #Fshd = fraccion de hueco sombreada por voladizos (para calcular PXI)  
            #Un lucernario no tiene voladizos:  
            SLF=0  
            #<-----Fshd=0  
  
        # si no es lucernario---> ventana---> puede tener voladizos  
        else:  
            #ASHRAE: Tabla 12 (pag 17.9)  
            #en funcion de la orientacion y la latitud se obtiene un factor, SLF:  
            SLF=tabla12 (b[i] [6], c [10])    ## (orientacion, latitud)  
  
        #si en el textCtrl: voladizo L, hay un valor, asigno ese valor a una variable  
        if type(self.parent.panelEnvolvente.ventanas[i].elementosProteccionSolar[0])==unicode:
```

```
voladizo_L = self.parent.panelEnvolvente.ventanas[i].elementosProteccionSolar [0]
#Voladizo L (en CEX) = Doh (en ASHRAE)

else:
    #si en el textCtrl: Retranqueo, hay un valor, asigno ese valor a una variable
    if type(self.parent.panelEnvolvente.ventanas[i].elementosProteccionSolar[5])==unicode:
        voladizo_L = self.parent.panelEnvolvente.ventanas[i].elementosProteccionSolar [5]
        #la parte de arriba de un retranqueo es como un pequeño voladizo
    else:
        #no hay ni voladizo ni retranqueo-----> =0
        voladizo_L = 0

#si en el textCtrl: voladizo D, hay un valor, asigno ese valor a una variable
if type (self.parent.panelEnvolvente.ventanas[i].elementosProteccionSolar[1])==unicode:
    voladizo_D = self.parent.panelEnvolvente.ventanas[i].elementosProteccionSolar [1]
    #Voladizo D (en CEX) = Xoh (en ASHRAE)
else:
    voladizo_D = 0

#Para calcular Fshd necesito la altura de la ventana. Tengo 2 opciones:

#1.El usuario introduce altura y longitud de la ventana
#si el valor de la altura (b[i] [9]) NO está vacío,
if b[i][9]!="":
    #hago los calculos normalmente
    Q = (float (SLF) * float (voladizo_L) - float (voladizo_D)) / float (b[i] [9])

#2.El usuario sólo introduce la superficie de la ventana y no sé el valor de la altura
else:
    #aproximo el valor de la altura como la raiz cuadrada de la superficie:(b[i][2])
    Q = (float (SLF) * float (voladizo_L) - float (voladizo_D)) / (math.sqrt (float (b[i] [2])))

#Q y QQ son valores intermedios inexistentes que me sirven para calcular Fshd

if Q>0:
    QQ = Q
else:
    QQ = 0

if QQ>1:
    Fshd = 1
else:
    Fshd = QQ

# Q = (SLF * Voladizo_L - Voladizo_D) / altura ventana
# QQ = max (0, Q)
# Fshd = min (1, QQ)

# Fshd = min (1, max (0, ((SLF*Voladizo_L - Voladizo_D)/alturaVentana))) –ecuacion completa
```

```

#PXI = Peak Exterior Irradiance (W/m2)
#factor para calcular CFfen
#ASHRAE: Tabla10 (pag 17.9)
Irradiancia = tabla10 (b[i] [6], c [10])    # (orientacion, latitud)
ED = Irradiancia [0]                       #ED = Direct irradiance (W/m2)
Ed = Irradiancia [1]                       #Ed = diffuse irradiance (W/m2)
Et = Irradiancia [2]                       #Et = total irradiance (W/m2)

#si el hueco NO esta sombreado ----> soleado ----> Fshd=0 and orientacion diferente de Norte
if Fshd == 0 and b[i][6]!='Norte':
    #PXI = Tx * Et (donde Tx pongo 1 por el momento)
    PXI = 1 * Et
#si el hueco esta sombreado ----> Fshd!=0 or orientacion = Norte
else:
    #PXI=Tx*(Ed + (1-Fshd)*ED) -->si en esta ecuacion Fshd=0 se tiene la de arriba porque Et
    =ED+Ed
    PXI = 1 *(Ed + (1-Fshd)*ED)

#SHGC = Solar Heat Gain Coefficient
#factor para calcular CFfen
#F = factor solar modificado = Fs * ((1-FM)*g + FM*0.04*Um*alfa)
#Fs=Factor de sombra-->si no se justifica, Fs=1
#FM=fraccion de hueco ocupada por marco = (b[i] [7])/100
#g=factor solar = b[i] [4]
#UM=transmitancia termica del marco = b[i] [5]
#alfa =absortividad del marco = b[i] [10]
SHGC = 1 * ( (1-float(b[i][7])/100) * float(b[i][4]) + float(b[i][7]) /100 *0.04 *float(b[i][5])
*float(b[i][10]) )

#IAC = Interior Attenuation Coefficient
#factor para calcular CFfen
#ASHRAE: Ecuacion 29 (pag 17.10)
#IAC = 1 + Fcl*(IACcl - 1)
#Fcl = shade fraction closed [0 to 1]
#IACcl = interior attenuation coefficient of fully closed configuration (tabla 14 (pag 17.10))
IAC = 1

#FFs = Fenestration Solar Load Factors
#factor para calcular CFfen
#ASHRAE: tabla 13 (pag17.10)
FFs = tabla13(c [6], b[i] [6])    ## (tipo de edificio, orientacion)

#CFfen = U· (dT - 0.46*DR) + PXI*SHGC*IAC*FFs
#ASHRAE: Ecuacion 25 (pag 17.9)
#dT = salto termico refrigeracion= Temp diseño refrigeracion - 24
#DR = daily range = Oscilacion Media Diaria
CFfen = float (b[i] [3]) * (float(c [2])-24.0 - 0.46*c [4]) + PXI*SHGC*IAC*FFs

```

```
#potHuecoRef = A * CFfen
#ASHRAE: Ecuacion 23 (pag 17.9)
potHuecoRef = float(b[i][2]) * CFfen
print 'potHuecoRef',potHuecoRef

#potHuecosTotalesRef = suma de la potHuecoRef de todos los huecos
potHuecosTotalesRef += potHuecoRef

#potHuecoCal = A * U * dT
#dT = salto termico calefaccion = 20 - Temp diseño calefaccion
potHuecoCal = float (b[i] [2]) * float (b[i] [3]) * (20 - float(c [3]))

#potHuecosTotalesCal = suma de la potHuecoCal de todos los huecos
potHuecosTotalesCal += potHuecoCal

#potEnvolvente = suma de las potencias de superficiesOpacas, Puertas y Ventanas (y PT)
potEnvolventeRef = potSupOpacasRef + potPuertasRef + potHuecosTotalesRef
potEnvolventeCal = potSupOpacasCal + potPuertasCal + potHuecosTotalesCal + potPT

## 2.-Cálculo de la potencia de Refrigeración/Calefacción por VentilaciónInfiltración:

#Para calcular el area expuesta al exterior:
areaExpuesta = 0
for i in self.parent.panelEnvolvente.cerramientos:
    #en la última posicion de los cerramientos:
    #- 'aire': en contacto con el aire, es cerramiento exterior
    #- 'vertical': es partición interior, tambien cuenta como area expuesta
    if i[-1]=='aire' or i[-1]=='vertical':
        areaExpuesta += float (i[2])

#Aul = unit leakage area (cm2/m2)
#ASHRAE: tabla 3 (pag 17.6)
# Es la superficie de fugas por unidad de superficie. Se utiliza para calcular AL
# Depende de la calidad de la construccion
Aul = tabla3a(c [5])          ## (año de construccion)

#AL = building effective leakage area at reference pressure difference = 4 Pa (cm2)
AL = areaExpuesta * Aul

#IDF = infiltration driving force (L/s·cm2)
#ASHRAE: Tabla 5 (pag 17.6)
IDFRef = tabla5(c [2], float(c [8]))    ## (temp diseño refrigeracion, altura de la planta)
IDFCal = tabla5(c [3], float(c [8]))    ## (temp diseño calefaccion, altura de la planta)

#Qi = Caudal infiltracion (L/s). 2 Formas según ASHRAE:
#1.ASHRAE: Ecuacion 8 (pag 17.5)
QiRef = AL * IDFRef
QiCal = AL * IDFCal
#2.ASRHAE: Ecuacion 6 (pag 17.5)
#Qi = Renovaciones/Hora * Volumen / 3.6 (L/s)
```

```
#Qv = Caudal ventilacion (L/s)
#ASHRAE: Ecuacion 11 (pag 17.7)
#Qv = 0.05 * superficieUtil + 3.5 (Nbr + 1)
#superficieUtil=c [7]
#Nbr +1 (Number bedrooms + 1) = Noc (Number occupants)
#supongo 3 personas cada 100m2 -----> Procedimientos alternativos
##    Qv = 0.05*float(c [7]) + 3.5*3*float(c [7])/100

#Qv = Renovaciones/hora * superficieUtil * alturaPlanta /3.6
#Se supone 1 ren/h
##    Qv = 1 * float(c [7]) * float(c [8]) / 3.6

#CTE:
#Qmin = Numero de baños * 15 l/s + Numero de cocinas * 50 l/s    (l/s)
Qv = float (e [0]) * 15 + float (e [1]) * 50

#Qvi = Caudal combinado ventilacion/infiltracion (L/s)
QviRef = Qv + QiRef
QviCal = Qv + QiCal
##    Qvi = Qv + Qi

#Potencia VentilacionInfiltracion (sensible)
#ASHRAE: Ecuacion 15 (Pag 17.7)
#qvi,s = Cs * Qvi * dT
#Cs = 1.23
#dT = salto termico
potVentilacionInfiltracionRef = 1.23 * QviRef * (c[2]-24.0)
potVentilacionInfiltracionCal = 1.23 * QviCal * (20 - c[3])
##    potVentilacionInfiltracionRef = 1.23 * Qvi * (c[2]-24.0)
##    potVentilacionInfiltracionCal = 1.23 * Qvi * (20 - c [3])

## 3.-Cálculo de la potencia de Refrigeración por ganancias internas:

#Potencia Ganancia Interna (sensible)
#ASHRAE: Ecuacion 30 (pag 17.10)
#qig,s = 136 + 2.2 * SuperficieUtil + 22 * Noc
#Superficie Util=c [7]
#Noc (Number occupants) = Nbr +1 (Number bedrooms + 1)
#supongo 3 personas cada 100m2
potGananciaInternaRef = 136 + 2.2 * float(c[7]) + 22 *3*float(c[7])/100

## 4.-Sumatorio de potencias debidas a la Envolvente, VentilaciónInfiltración (y las ganancias internas)
potSubtotalRef = potEnvolventeRef + potVentilacionInfiltracionRef + potGananciaInternaRef
potSubtotalCal = potEnvolventeCal + potVentilacionInfiltracionCal
```

5.-Cálculo de la potencia de Refrigeración/Calefacción por pérdidas de distribución:

```
#Fdl = duct loss/gain factor
##ASHRAE: tabla 11 (pag 17.8)
#es un factor para calcular las perdidas de distribucion
#qd (Cal/Ref) = Fdl * potSubtotal (Cal/Ref)
#ASHRAE: Ecuacion 19 (pag 17.8)
## potPerdidasDistribucionRef = FdlRef * potSubtotalRef
## potPerdidasDistribucionCal = FdlCal * potSubtotalCal

## potRefrigeracionSensibleTotal = potSubtotalRef + potPerdidasDistribucionRef
## potCalefaccionSensibleTotal = potSubtotalCal + potPerdidasDistribucionCal

## potFrio = potRefrigeracionSensibleTotal
## potCalor = potCalefaccionSensibleTotal

potFrio = potSubtotalRef
potCalor = potSubtotalCal

wx.MessageBox('Potencia Calor '+str(round(potCalor/1000,2))+'(kW): \nPotencia Frio
'+str(round(potFrio/1000,2))+' (kW): ', 'Residential Load Calculations')
```

Carga latente

1.-Cálculo de la potencia de Refrigeración por VentilaciónInfiltración:

```
#potVentilacionInfiltracion (Latente)
#ASHRAE: Ecuacion 16 (pag 17.7)
#qvi,l = CI * Qvi * dW
#CI = 3010 W / (L·s)
#dW = diferencia de humedad interior/exterior
## potVentilacionInfiltracionLatente = 3010 * Qvi * dW
```

2.-Cálculo de la potencia de Refrigeración por ganancias internas:

```
#Potencia Ganancia Interna (latente)
#ASHRAE: Ecuacion
#qig,l = 20 + 0.22 * SuperficieUtil + 12 * Noc
#Superficie Util=c [7]
#Noc (Number occupants) = Nbr +1 (Number bedrooms + 1)
#supongo 3 personas cada 100m2
## potGananciaInternaLatente = 20 + 0.22 * float(c [7]) + 12*3*float(c[7])/100

## potRefrigeracionLatenteTotal = potVentilacionInfiltracionLatente + potGananciaInternaLatente
```

#####

```
def coggerPT(self):
    PTparaASHRAE = []
    for i in self.parent.panelEnvolvente.puentesTermicos:
        PTparaASHRAE.append([i [0], i [3], i [4]])
    return PTparaASHRAE
```



```
def cogerCerramientos(self):                                ##los cerramientos son arrays dentro de array
    cerramientosParaAshrae = []
    for i in self.parent.panelEnvolvente.cerramientos:
        cerramientosParaAshrae.append([i[0],i[1],i[2],i[3],i[5],i[-1]])
#nombre(a[i][0]),tipo(a[i][1]),A(a[i][2]),U(a[i][3]),Orientacion(a[i][5]),contacto
    return cerramientosParaAshrae

def cogerVentanas(self):                                    ##las ventanas son objetos dentro de un array. Diferente escritura
    ventanasParaAshrae=[]
    for j in self.parent.panelEnvolvente.ventanas:

ventanasParaAshrae.append([j.descripcion,j.tipo,j.superficie,j.Uvidrio,j.Gvidrio,j.Umarco,j.orientacion,j.p
orcMarco,j.tieneProteccionSolar,j.altura,j.absortividadValor,j.cerramientoAsociado])
    return ventanasParaAshrae

def datos(self):
    datosNecesarios=[]

##    provincia    c[0]
    provincia=self.parent.panelDatosGenerales.provinciaChoice.GetStringSelection()
    datosNecesarios.append (provincia)

##    ciudad    c[1]
    ciudad=self.parent.panelDatosGenerales.localidadChoice.GetStringSelection()
    datosNecesarios.append (ciudad)

##    temperaturas
    temperaturas=temperaturasDiseno(provincia,ciudad)
    tRefrigeracion=temperaturas[0]                # c [2]
    tCalefaccion=temperaturas[1]                  # c [3]
    Oscilacion=temperaturas [2]                   # c [4]
    datosNecesarios.append (tRefrigeracion)
    datosNecesarios.append (tCalefaccion)
    datosNecesarios.append (Oscilacion)

##    anoConstruccion    c[5]
    anoConstruccion=self.parent.panelDatosGenerales.anoConstruccionChoice.GetStringSelection()
    datosNecesarios.append (anoConstruccion)

##    tipoEdificio    c[6]
    tipoEdificio=self.parent.panelDatosGenerales.tipoEdificioChoice.GetStringSelection()
    datosNecesarios.append (tipoEdificio)

##    superfUtil    c[7]
    superfUtil=self.parent.panelDatosGenerales.superficie.GetValue()
    datosNecesarios.append (superfUtil)

##    alturaPlanta    c[8]
    alturaPlanta=self.parent.panelDatosGenerales.alturaMediaLibre.GetValue()
    datosNecesarios.append (alturaPlanta)
```

```
##      masaParticiones  c[9]
masaParticiones=self.parent.panelDatosGenerales.masaParticiones.GetStringSelection()
datosNecesarios.append (masaParticiones)

##      geografia
geografia=temperaturasDiseno(provincia,ciudad)
latitud=geografia[3]                                #c[10]
longitud=geografia[4]                                #c[11]
datosNecesarios.append (latitud)
datosNecesarios.append (longitud)

return datosNecesarios

def cogerDatos(self):
    a = []
    a.append(self.banos.GetValue())
    a.append(self.cocinas.GetValue())
    a.append(self.ocupacion.GetValue())
    return a
```

5.1.3. ASHRAETerciario.py:

```
# -*- coding: cp1252 -*-
#Boa: Dialog: Dialog1

import wx
from TablasASHRAE import *
from math import *

    ##si importo así, tengo todas las funciones de
    ## TablasASHRAE disponibles.Solo con poner su nombre me basta

    ##Si pusiera import TablasASHRAE
    ##luego tendría que poner: TablasASHRAE.nombreFuncion

def create(parent):
    return Dialog1(parent)

[wxID_DIALOG1, wxID_DIALOG1CALCULARBUTTON,
wxID_DIALOG1CALCULARTERCIARIOBUTTON, wxID_DIALOG1OCUPACIONTEXT,
wxID_PANEL1ACCESORIOTEXT, wxID_PANEL1ACCESORIOCHOICE,
wxID_PANEL1TITULOTXT, wxID_BANOSTEXT, wxID_DIALOG1BANOS,
wxID_COCINASTEXT, wxID_DIALOG1COCINAS, wxID_DIALOG1DEINICIOTEXT,
wxID_DIALOG1HORARIOINICIOCHOICE, wxID_DIALOG1HORARIOFIN,
wxID_DIALOG1ATEXT, wxID_DIALOG1HORASFINTEXT, wxID_DIALOG1PERSONASTEXT,
wxID_DIALOG1OCUPACION, wxID_DIALOG1POTENCIATEXT, wxID_DIALOG1POTENCIA,
wxID_DIALOG1WTEXT, wxID_DIALOG1HORARIOTEXT, wxID_DIALOG1HORASINICIOTEXT
] = [wx.NewId () for _init_ctrls in range (23)]

class Dialog1(wx.Dialog):

    def _init_ctrls(self, prnt):

        wx.Dialog.__init__(self, id=wxID_DIALOG1, name="", parent=prnt,
            pos=wx.Point(270, 100), size=wx.Size(501, 350),
            style=wx.DEFAULT_DIALOG_STYLE, title='Nonresidential Load Calculations')
        self.SetClientSize (wx.Size (485, 350))
        self.SetBackgroundColour ('white')

        self.tituloTxt = wx.StaticText (id=wxID_PANEL1TITULOTXT,
            label=u'Cálculo de la potencia de Calefacción/Refrigeración', name=u'tituloTxt',
            parent=self, pos=wx.Point(25, 24), size=wx.Size(153, 16),
            style=0)
        self.tituloTxt.SetFont (wx.Font (12, wx.SWISS, wx.ITALIC, wx.BOLD,
            False, u'Georgia'))
        self.tituloTxt.SetForegroundColour (wx.Colour (0, 64, 128))

        self.calcularTerciarioButton = wx.Button(id=wxID_DIALOG1CALCULARTERCIARIOBUTTON,
            label=u'Calcular', name=u'calcularTerciarioButton', parent=self,
            pos=wx.Point(188, 260), size=wx.Size(100, 32), style=0)
        self.calcularTerciarioButton.Bind (wx.EVT_BUTTON, self.onCalcularTerciario,
            id=wxID_DIALOG1CALCULARTERCIARIOBUTTON)

        self.ocupacionText = wx.StaticText(id=wxID_DIALOG1OCUPACIONTEXT,
            label=u'Ocupaci\x3f n :', name=u'ocupacionText', parent=self,
            pos=wx.Point(40, 86), size=wx.Size(60, 13), style=0)
```

```
self.ocupacion = wx.TextCtrl(id=wxID_DIALOG1OCUPACION,
                             name=u'ocupacion', parent=self, pos=wx.Point(220, 82),
                             size=wx.Size(44, 20), style=0, value=u")

self.personasText = wx.StaticText(id=wxID_DIALOG1PERSONASTEXT,
                                  label=u'personas', name=u'personasText', parent=self,
                                  pos=wx.Point(274, 86), size=wx.Size(50, 13), style=0)

self.horarioText = wx.StaticText(id=wxID_DIALOG1HORARIOTEXT,
                                  label=u'Horario de funcionamiento :', name=u'horarioText', parent=self,
                                  pos=wx.Point(40, 126), size=wx.Size(135, 13), style=0)

self.deInicioText = wx.StaticText(id=wxID_DIALOG1DEINICIOTEXT,
                                   label=u'de', name=u'deInicioText', parent=self,
                                   pos=wx.Point(190, 126), size=wx.Size(13, 13), style=0)

self.horarioInicioChoice = wx.Choice(choices=['1', '2',
                                              '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18',
                                              '19', '20', '21', '22', '23', '24'],
                                     id=wxID_DIALOG1HORARIOINICIOCHOICE, name=u'horarioInicioChoice',
                                     parent=self, pos=wx.Point(220, 124), size=wx.Size(44, 30),
                                     style=0)
self.horarioInicioChoice.Bind(wx.EVT_CHOICE, self.OnHorarioInicioChoice,
                              id=wxID_DIALOG1HORARIOINICIOCHOICE)

self.horasInicioText = wx.StaticText(id=wxID_DIALOG1HORASINICIOTEXT,
                                      label=u'horas', name=u'horasInicioText', parent=self,
                                      pos=wx.Point(274, 126), size=wx.Size(50, 13), style=0)

self.aText = wx.StaticText(id=wxID_DIALOG1ATEXT,
                           label=u'a', name=u'aText', parent=self,
                           pos=wx.Point(190, 154), size=wx.Size(10, 13), style=0)

self.horarioFin = wx.TextCtrl(id=wxID_DIALOG1HORARIOFIN,
                              name=u'horarioFin', parent=self, pos=wx.Point(220, 152),
                              size=wx.Size(44, 20), style=0, value=u")
self.horarioFin.Enable(False)

self.horasFinText = wx.StaticText(id=wxID_DIALOG1HORASFINTEXT,
                                  label=u'horas', name=u'horasFinText', parent=self,
                                  pos=wx.Point(274, 154), size=wx.Size(50, 13), style=0)

self.potenciaText = wx.StaticText(id=wxID_DIALOG1POTENCIATEXT,
                                   label=u'Potencia de los aparatos :', name=u'potenciaText', parent=self,
                                   pos=wx.Point(40, 200), size=wx.Size(130, 13), style=0)

self.potencia = wx.TextCtrl(id=wxID_DIALOG1POTENCIA,
                            name=u'potencia', parent=self, pos=wx.Point(220, 198),
                            size=wx.Size(44, 20), style=0, value=u")

self.wText = wx.StaticText(id=wxID_DIALOG1WTEXT,
                           label=u'W', name=u'wText', parent=self,
                           pos=wx.Point(274, 200), size=wx.Size(50, 13), style=0)

def __init__(self, parent):
    self.parent = parent
    self._init_ctrls(parent)
```

#####

```
# CÁLCULO DE CARGAS DE CALEFACCIÓN Y REFRIGERACIÓN
# Según ASHRAE: 2009 Handbook - Fundamentals
# Todas las tablas utilizadas se encuentran en: Plugins\betacomb\TablasASHRAE.py
```

```
### Ref = Refrigeración
### Cal = Calefacción
```

Nonresidential cooling and heating load calculations (Chapter 18)

#Cuando se pulsa el botón Calcular, llama a esta función:

```
def onCalcularTerciario(self,event):
    self.calculosTerciario ()
```

#que a su vez llama a la siguiente, la cual recoge todos los datos que hacen falta para realizar los calculos:

```
def calculosTerciario(self):
    a = self.cogerCerramientos ()
```

```
b = self.cogerVentanas ()
```

#En datos se encuentran datos importantes para la realizacion de los calculos

```
c = self.datos ()
```

```
d = self.cogerPT ()
```

#aqui se encuentran los datos introducidos en la ventana por el usuario

```
e = self.cogerDatosTerciario ()
```

----- Potencia debida a la OCUPACIÓN -----

#En función de la actividad realizada se liberará una cantidad de calor sensible/latente diferente

#Se ha diferenciado según la actividad seleccionada en el panel Instalaciones--> iluminación

#TablasASHRAE: def actividad (p. ej.: 'Administrativo en general')

```
p = actividad (e [4])
```

#P=Potencia por ocupacion (W) = personas * Potencia (sensible) liberada/persona

#personas=introducido por el usuario en la ventana

```
P = float (e [0]) * p [0]
```

#fin de la ocupación = horarioInicio (choice de la ventana) + horas de uso

```
finOcupacion = float(e[2]) + float(e[5])
```

#si finOcupación > 24h. ---> no puede ser una hora mayor que 24

```
if finOcupacion>24:
```

#entonces se le resta 24. Si nos salía 28(p.ej.) será 4--> 4 de la mañana cuando cierra el edificio

```
finOcupacion=finOcupacion-24
```

```
#Horario = horario de uso del edificio
#TablasASHRAE: def horarioFuncionamiento (horarioInicio (choice), horarioFin)
Horario = horarioFuncionamiento (float (e [2]), finOcupacion)
#horarioOcupacion es la potencia, hora a hora, debida a la ocupación
horarioOcupacion=[]
for i in range(24):
    q= P * Horario[i]
    horarioOcupacion.append (q)

#TablasASHRAE: def actividad
porcentajeConvectiva=p[1]

Q= []
#Se aplica RTS para saber la carga de refrigeracion radiante
#ASHRAE: tabla 19 (pag. 18.29)
#c [9]=masa de las particiones
k = NonsolarRTS(c [9])

for i in range(24):
    x=0
    contador = i
    contadorFinal = contador -23

    Qr=0
    while contador >= contadorFinal and x<24:
        q=horarioOcupacion [contador]*k[x]*(1 - porcentajeConvectiva)
        Qr+=q
        if contador ==0:
            contador=23
        else:
            contador = contador -1
        x = x+1
    Q.append (Qr)

#en potOcupacionPorHoras, se guarda la carga de Ref debida a la ocupacion, hora a hora
potOcupacionPorHoras=[]
for i in range(24):
    Ocupacion = horarioOcupacion[i]*porcentajeConvectiva + Q[i]
    potOcupacionPorHoras.append (Ocupacion)

#potRefLatente = personas * Potencia (latente) liberada/persona
potRefLatente = float(e[0]) * p[2]
```

----- Potencia debida a la ILUMINACIÓN -----

```
#P=Potencia de iluminacion (W), introducida por el usuario en el panel Instalaciones --> iluminacion
P=float (e [3])
```

```
#Es el mismo método que para ocupación
```

```
horarioIluminacion=[]
for i in range(24):
    q= P * Horario[i]
    horarioIluminacion.append (q)

porcentajeConvectiva=0.52

Q=[]
k = NonsolarRTS(c [9])
for i in range(24):
    x=0
    contador = i
    contadorFinal = contador -23

    Qr=0
    while contador >= contadorFinal and x<24:
        q=horarioIluminacion[contador]*k[x]*(1 - porcentajeConvectiva)
        Qr+=q
        if contador ==0:
            contador=23
        else:
            contador = contador -1
        x = x+1
    Q.append(Qr)

potIluminacionPorHoras=[]
for i in range(24):
    Iluminacion = horarioIluminacion[i]*porcentajeConvectiva + Q[i]
    potIluminacionPorHoras.append (Iluminacion)
```

----- Potencia debida a los EQUIPOS -----

```
#G=Ganancia de calor de los equipos (W)
#Supongo que el 50% de la potencia de los equipos pasa a ser ganancia de calor (es conservador)
#e [1]=potencia de los equipos (introducida por el usuario en la ventana)
G= 0.5 * float (e [1])
```

```
#Es el mismo método que para ocupación
```

```
horarioEquipos=[]
for i in range(24):
    q= G * Horario[i]
    horarioEquipos.append (q)
```



```
porcentajeConvectiva=0.6

Q= []
k = NonsolarRTS(c [9])
for i in range(24):
    x=0
    contador = i
    contadorFinal = contador -23

    Qr=0
    while contador >= contadorFinal and x<24:
        q=horarioEquipos [contador]*k[x]*(1 - porcentajeConvectiva)
        Qr+=q
        if contador ==0:
            contador=23
        else:
            contador = contador -1
        x = x+1
    Q.append (Qr)

potEquiposPorHoras=[]
for i in range(24):
    equipos = horarioEquipos[i]*porcentajeConvectiva + Q[i]
    potEquiposPorHoras.append (equipos)
```

----- Potencia debida a los PUENTES TÉRMICOS (sólo calefacción) -----

```
potPT=0
#recorro los PT:
for i in range(len(self.parent.panelEnvolvente.puentesTermicos)):
    #Para cada PT:
    #->PT = fi * L * dT (W)
    #fi=conductividad lineal
    #L=longitud del PT
    #dT=salto termico calefaccion= 22.2 - Temp Diseño Calefaccion
    PT = float (d[i] [1]) * float (d[i] [2]) * (22.2 - float(c [3]))
    #->suma de cada PT
    potPT += PT
```

----- Potencia debida a los CERRAMIENTOS -----

```
potRefrigeracionMeses=[]

#en refrigeración, se realizan los cálculos para los meses de julio, agosto y septiembre
meses=['julio','agosto','septiembre']

#ASHRAE Capítulo 14, Tabla 1/2 (pag. 14.2/14.7)
for mes in meses:
```

```
if mes=='julio':
    #ET = Equation of Time (min)
    ET=-6.4
    #angDelta = Declination (grados)
    angDelta=20.4
    #Eo = Extraterrestrial radiant flux (W/m2)
    Eo=1324
    #taub = Clear sky optical depth for beam irradiance
    taub=0.556
    #taud = Clear sky optical depth for diffuse irradiance
    taud=1.779
    #tempPorHoras = temperatura hora a hora de un día tipo del mes
    tempPorHoras=c[12]
elif mes=='agosto':
    ET=-3.6
    angDelta=11.8
    Eo=1336
    taub=0.593
    taud=1.679
    tempPorHoras=c[13]
else:
    ET=6.9
    angDelta=-0.2
    Eo=1357
    taub=0.431
    taud=2.151
    tempPorHoras=c[14]

potVentanas=0
potSupOpacasCal=0
envolventeCal=0
#en envolvente guardo arrays de potRef por horas de cada cerramiento, para luego poder sumarlos
envolvente=[]
#en ventanas guardo los arrays de potRef por horas de cada ventana, para luego poder sumarlas
ventanas=[]

#Para cada cerramiento se realiza lo siguiente:
for i in range(len(self.parent.panelEnvolvente.cerramientos)):

    sumado=False

    #si el cerramiento es techo o suelo:
    if a[i][5]=='Techo' or a[i][5]=='Suelo':
        angInclinacion=0
    else:
        #medianera---> no me interesan los calculos siguientes, potRef=0---> pongo un valor
        cualquiera, (p.e.50)
        if a[i][-1]=='edificio':
            angInclinacion=50
            angOrientacion=50
        else:
            #particion interior, potRef=0 ---> pongo un valor cualquiera, (p.e.50)
            if a[i][1]=='Partición Interior':
                angInclinacion=90
                angOrientacion=50
```

```
else:
    #si es muro:
    angInclinacion=90
    #ese muro tendrá una orientacion
    #TablasASHRAE: def orientacion (orientacion del muro)
    angOrientacion=orientacion(a[i] [5])

#c [0]=provincia
if c[0]=='Las Palmas':
    LSM=15    #Local Standard Meridian (grados) ----> Canarias tiene diferente zona horaria
else:
    LSM=30

#Estos arrays contendrán valores hora a hora
AST=[]
angH=[]
angBeta=[]
angFi=[]
angGamma=[]
angTita=[]
m=[]
Eb=[]
Eth=[]
Y=[]
Ed=[]
Etd=[]
Etr=[]
Et=[]

solAirTemp=[]
heatInput=[]

#Para cada hora tengo que hacer lo siguiente:

for j in range(1,25):
    #ASHRAE Capítulo 14, Ecuación 7 (pag. 14.7)
    #AST=Aparent Solar Time
    #c [11]=longitud (de la ciudad)
    Ast=j + ET/60 + (float(c [11]) - LSM)/15
    AST.append (ast)

    #ASHRAE Capítulo 14, Ecuación 11 (pag. 14.8)
    #angH=angulo horario (grados): =0(mediodia). <0(mañana). >0(tarde)
    H= 15 * (ast - 12)
    angH.append (H)

    #ASHRAE Capítulo 14, Ecuación 12 (pag. 14.8)
    #angBeta=altitud solar (grados)
    #c [10]=latitud de la ciudad
    Bet= asin (cos(c [10]*pi/180)*cos (angDelta*pi/180)*cos (H*pi/180)+sin(c [10]*pi/180)* sin
(angDelta*pi/180))
    #En python en radianes. Para transformarlo a grados:
    Beta=Bet*180/pi
```

```
angBeta.append (Beta)

#angFi=acimut solar (grados)
fi= acos ((sin(Beta*pi/180)*sin(c[10]*pi/180)-
sin(angDelta*pi/180))/(cos(Beta*pi/180)*cos(c[10]*pi/180)))
Fi=fi*180/pi

if H<0 and Fi<0:
    pass
elif H>0 and Fi>0:
    pass
elif H<0 and Fi>0:
    Fi= -1 * Fi
elif H>0 and Fi<0:
    Fi= -1 * Fi

angFi.append (Fi)

#ASHRAE Capítulo 14, Ecuación 21 (pag. 14.10)
#angGamma=surface-solar azimuth (grados)
if angInclinacion==0: #si es techo o suelo ---> Gamma no me interesa y vale, p.ej. 22
    Gamma=22
    angGamma.append (Gamma)

else:
    Gamma = float (Fi) - float (angOrientacion)
    angGamma.append (Gamma)

#ASHRAE Capítulo 14, Ecuación 22 (pag. 14.10)
#angTita=angulo de incidencia (grados)
tita = acos (cos (Beta*pi/180)*cos (Gamma*pi/180)*sin(angInclinacion*pi/180)
+ sin (Beta*pi/180)*cos (angInclinacion*pi/180))
Tita = tita*180/pi
angTita.append (Tita)

#ASHRAE Capítulo 14, Ecuación 19/20 (pag. 14.9)
#ab=beam air mass exponent
ab = 1.219 - 0.043*taub - 0.151*taud - 0.204*taub*taud
#ad=diffuse air mass exponent
ad = 0.202 + 0.852*taub - 0.007*taud - 0.357*taub*taud

if Beta<0:
    M=0
    eb=0
    ed=0
else:
    #ASHRAE Capítulo 14, Ecuación 16/17/18 (pag. 14.9)
    M = 1/(sin(Beta*pi/180)+0.50572*(6.07995+Beta)**(-1.6364))
    eb = Eo*exp (-taub*M**ab)
    ed = Eo*exp (-taud*M**ad)
```

```
#m=relative air mass
#m=ratio of the mass of atmosphere in the actual earth/sun path to the mass that would exist
if the sun were directly overhead.
m.append (M)
#Eb=beam normal irradiance (irradiancia normal directa-medida perpendicularmente a los
rayos del sol-) W/m2
Eb.append (eb)
#Ed=diffuse horizontal irradiance (irradiancia horizontal difusa -medida en la superficie
horizontal-) W/m2
Ed.append(ed)

if cos (Tita*pi/180)<0:
    etb = 0
else:
    etb=eb*cos(Tita*pi/180)
#Etb=surface beam irradiance (W/m2)
#ASHRAE Capítulo 14, Ecuación 26 (pag. 14.10)
Etb.append (etb)

#Y=ratio of sky diffuse radiation on vertical surface to sky diffuse radiation on horizontal
surface
#ASHRAE Capítulo 14, Ecuación 28 (pag. 14.10)
if cos(Tita*pi/180)> -0.2:
    y = 0.55 + 0.437*cos (Tita*pi/180) + 0.313*cos(Tita*pi/180)*cos(Tita*pi/180)
    Y.append(y)
else:
    y = 0.45
    Y.append(y)

#Etd=diffuse irradiance (W/m2) --> vertical surfaces
#ASHRAE Capítulo 14, Ecuación 27 (pag. 14.10)
etd = ed * y
Etd.append (etd)

#Etr=ground reflected irradiance (W/m2)
#ASHRAE Capítulo 14, Ecuación 31 (pag. 14.10)
etr = (eb*sin(Beta*pi/180) + ed)*0.2*(1 - cos(angInclinacion*pi/180))/2
Etr.append (etr)

#Et=total surface irradiance (W/m2)
#ASHRAE Capítulo 14, Ecuación 25 (pag. 14.10)
et = etd + etb + etr
Et.append (et)
```

#sol-Air Temperature is the outdoor air temperature that, in the absence of all radiation changes gives
#the same rate of heat entry into the surface as would give the combination of incident solar radiation,
#radiant energy exchange with the sky and other outdoor surroundings, and convective heat exchange
with outdoor air.

$$\#solairT = outdoorT + \alpha * Et / ho - \epsilon * dR / ho$$

#ASHRAE: Capítulo 18, Ecuación 30 (pag. 18.22)

#alfa=solar absorptance

#Et=total solar radiation incident on surface (W/m2)

#ho=coefficient of heat transfer by longwave radiation and convection at outer surface

(W/m²·K)

#alfa/ho=r

r=0.053

#epsilon=hemispherical emittance of surface

#dR=difference between long-wave radiation incident on surface from sky and surroundings

and emitted by

#blackbody at outdoor air temperature (W/m2)

#epsilon*dR/ho=s

```
if a[i][5]=="Techo" or a[i][5]=="Suelo": #a[i][5]=orientacion del cerramiento
```

 $s=4$

else:

 $s=0$
$$\text{solair} = \text{tempPorHoras} [\text{j}-1] + \text{r}^*(\text{et}) - \text{s}$$

```
solAirTemp.append (solair)
```

$$\#q = U * A * (\text{solairT} - \text{temperaturaInteriorEnRefrigeracion})$$

#ASHRAE: ecuacion 31 (pag. 18.23)

#Con esta ecuacion se obtiene el calor que entra a traves del cerramiento

```
q = float(a[i][2]) * float(a[i][3]) * float(solair - 23.9)
```

```
heatInput.append (q)
```

Conduction Time Series

#en funcion de las características del cerramiento se obtienen los Conduction Time Factors (CTF)

#y con ellos (y junto con el calor que entra) se obtiene la ganancia de calor del cerramiento

```
if a[i][5]=='Techo':
```

```
k1 = roofCTS (a[i] [4], a[i] [3])
```

porcentajeConvectiva=0.4

```
elif a[i][1]=='Fachada':
```

```
k1 = wallCTS(a[i][4],a[i][3])
```

porcentajeConvectiva=0.54

```
elif a[i][5]=='Suelo' or a[i][1]==u'Partición Interior':
```

$$\mathbf{k1} = [0,0]$$

porcentajeConvectiva=0

$$Q \equiv \square$$

```
for g in range(24):
```

 $x=0$

```

contador = g
contadorFinal = contador -23
QQ=0
while contador >= contadorFinal and x<24:
    qq = heatInput[contador] * k1[x]
    QQ += qq
    if contador ==0:
        contador=23
    else:
        contador = contador -1
    x = x+1

```

```
Q.append (QQ)
```

Radiant Time Series

#La ganancia de calor se divide en parte convectiva y parte radiante:
 #a la parte radiante se le aplica RTS: en funcion del cerramiento se obtiene unos RTF
 #y se obtiene la potencia de refrigeracion radiante, Qradiant (W)
 k2=NonsolarRTS(c [9])

#potRefrigeracion radiante de un cerramiento, por horas (W)

```

Qradiant= []
for g in range(24):
    x=0
    contador = g
    contadorFinal = contador -23
    QQQ=0
    while contador >= contadorFinal and x<24:
        qq = Q[contador] * k2[x] * (1 - porcentajeConvectiva)
        QQQ += qq
        if contador ==0:
            contador=23
        else:
            contador = contador -1
        x = x+1

```

```
Qradiant.append (QQQ)
```

#potRefrigeracion convectiva de un cerramiento, por horas (W)

```

Qconvective= []
for j in range(24):
    Qconvect = Q[j]* porcentajeConvectiva
    Qconvective.append (Qconvect)

```

#potencia de Refrigeracion TOTAL de un cerramiento, por horas (W) = parte convectiva +
 parte radiante

#si el cerramiento no está en contacto con el terreno, no es medianera

```

if a[i][-1]!='terreno' and a[i][-1]!='edificio':
    Qtotal= []
    for j in range(24):
        Q=Qradiant[j] + Qconvective[j]

```



```

Qtotal.append(Q)
else:
    #si el cerramiento esta en contacto con el terreno o es medianera, ---> potRef=0
    Qtotal=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

#voy añadiendo los arrays de la potRef de cada cerramiento para luego poder sumarlos
envolvente+= [Qtotal]

```

----- Potencia debida a los HUECOS -----

```
#Seguimos dentro del cerramiento anterior. Para cada hueco se realiza lo siguiente:
for v in range(len(self.parent.panelEnvolvente.ventanas)):

    #si cerramiento = cerramiento asociado a esta ventana(es decir, si esta ventana pertenece al
    cerramiento):
    #Se hace esto para mantener los valores de los calculos anteriores del cerramiento:
    if a[i][0]==b[v][-1]:

        #potSupOpacasCal (con ventana) = (Abruta - Aventana) * U * dT
        #dT=salto térmico
        xCal = (float(a[i][2]) - float(b[v][2])) * float(a[i][3]) * (22.2 - float(c[3]))

        #potCerramiento = suma de la potSupOpacasCal de cada cerramiento
        potSupOpacasCal += xCal

    #potVentanaCal = A * U * dT
    yCal = float (b[v] [2]) * float(b[v][3]) * (22.2 - float(c[3]))

    #potVentanas = suma de la potVentanaCal de cada ventana
    potVentanas += yCal

    #Si un cerramiento tiene una ventana,se cambia esta variable.
    #Indica que la potencia de calefaccion debida al propio cerramiento ya se ha sumado
    sumado=True
```

```
#se corresponde con un vidrio doble normal
#ASHRAE (type 5b) Capítulo 15, Tabla 10 (pag. 15.21)
SHG= [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.166, 0.321, 0.398, 0.438, 0.448, 0.449, 0.441,
0.403, 0.33, 0.185, 0, 0]
#Hemisf. diffuse (type 5b)
SHGCdif=0.60
#se supone que no hay persianas ---> IAC=1
IAC=[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]
beamSHG=[]
diffuseSHG=[]
conductionHG=[]
totalWindowHG=[]
```

```

for j in range(24):
    #ASHRAE: Capítulo 18, Ecuación 13/14/15/16 (pag. 18.14/18.15)
    #qb=beam solar heat gain = A * Etb * SHG (tita) * IAC (W)
    qb=float(b[v][2]) * Etb[j] * SHG[j] * IAC[j]
    beamSHG.append(qb)

    #qd=diffuse solar heat gain = A * (Etd + Er) * SHGCdif * IAC (W)
    qd=float(b[v][2]) * (Etd[j] + Etr[j]) * SHGCdif * IAC[j]
    diffuseSHG.append(qd)

    #qc=conduction heat gain = A * U * (Tªexterior por horas - Tªinterior) (W)
    qc=float(b[v][2]) * float(b[v][3]) * (tempPorHoras[j] - 23.9)
    conductionHG.append(qc)

    #totalWindowHG=total window heat gain = beam + diffuse + conduction
    totalWindow=qb + qd + qc
    totalWindowHG.append(totalWindow)

```

respecto las

```

#Se suponen las ventanas sin sombreado interior ---> para obtener mayor carga
#En este caso, la ganancia de calor solar directa debe ser tratada de forma separada

#ganancias de calor difuso y conductivo
#La ganancia de calor solar directa se trata como 100% radiante y se aplica SolarRTS
k3 = SolarRTS(c [9])

```

```

cooling=[]
for g in range(24):
    x=0
    contador = g
    contadorFinal = contador -23
    QQQ=0
    while contador >= contadorFinal and x<24:
        qq = beamSHG[contador] * k3[x]
        QQQ += qq
        if contador ==0:
            contador=23
        else:
            contador = contador -1
        x = x+1

    cooling.append(QQQ)

```

convectiva

```

#Las ganancias difusas y conductivas se suman y se dividen en parte radiante y parte
totalConvective=[]
totalRadiant=[]
for j in range(24):
    convective= (diffuseSHG[j] + conductionHG[j]) * 0.6
    radiant= (diffuseSHG[j] + conductionHG[j]) * (1 - 0.6)
    totalConvective.append(convective)
    totalRadiant.append(radiant)

```

#Se aplica NonsolarRTS a la parte radiante

```
radiant=[]
k4 = NonsolarRTS(c [9])
for g in range(24):
    x=0
    contador = g
    contadorFinal = contador -23
    QQQ=0
    while contador >= contadorFinal and x<24:
        qq = totalRadiant[contador] * k4[x]
        QQQ += qq
        if contador ==0:
            contador=23
        else:
            contador = contador -1
        x = x+1

    radiant.append(QQQ)
```

#Se suma la carga de refrigeracion (por ganancia directa)+parte convectiva+parte radiante

```
windowCoolingLoad=[]
for j in range(24):
    window=cooling[j] + totalConvective[j] + radiant[j]
    windowCoolingLoad.append(window)
```

#voy añadiendo los arrays de la potRef de cada ventana para luego poder sumarlas
ventanas+= [windowCoolingLoad]

no se tienen en cuenta las persianas ---> para obtener mayor carga

#Para los cerramientos que no tienen ventanas, todavia se debe sumar la potencia debida al propio cerramiento

```
if sumado !=True:
    #potCal = A * U * dT
    xCal = float(a[i][2]) * float(a[i][3]) * (22.2 - float(c[3]))

    #potSupOpacasCal = suma de la potCal de cada cerramiento
    potSupOpacasCal += xCal
else:
    pass
```

envolventeCal = potPT + potSupOpacasCal + potVentanas

INFILTRACIÓN

#Solo en calefacción. Se supone que en Ref, el edificio se mantiene en positive pressure y no existe

#Para calcular el area expuesta al exterior:

areaExpuesta = 0

for i in self.parent.panelEnvolvente.cerramientos:

#en la última posición de los cerramientos:

#- 'aire': en contacto con el aire, es cerramiento exterior

#- 'vertical': es partición interior, también cuenta como area expuesta

if i[-1]=='aire' or i[-1]=='vertical':

areaExpuesta += float(i[2])

#Aul = unit leakage area (cm²/m²)

#ASHRAE: Capítulo 17, Tabla 3 (pag 17.6)

Es la superficie de fugas por unidad de superficie. Se utiliza para calcular AL

Depende de la calidad de la construcción

Aul = tabla3a(c [5]) ## (año de construcción)

#AL = building effective leakage area at reference pressure difference = 4 Pa (cm²)

AL = areaExpuesta * Aul

#IDF = infiltration driving force (L/s·cm²)

#ASHRAE: Capítulo 17, Tabla 5 (pag 17.6)

IDFCal = tabla5(c [3], float (c [8])) ## (temp diseño calefacción, altura de la planta)

#Qi = Caudal infiltración (L/s).

#ASHRAE: Capítulo 17, Ecuación 8 (pag 17.5)

Qi = AL * IDFCal

VENTILACIÓN

#Potencia Ventilación Infiltración (sensible)

#ASHRAE: Ecuación 15 (Pag 17.7)

#q_{vi,s} = Cs * (Q_v+Q_i) * dT

#Cs = 1.23

#dT = salto termico

#c [7]=superficie util

#c [8]=altura de la planta

#c [3]=temperatura de diseño de calefacción

#l = 1 ren/h

potVentilacionInfiltracionCal = 1.23 * (1 * float(c[7]) * float(c[8]) / 3.6 + Qi) * (22.2 - float(c[3]))

#c [2]=temperatura de diseño de refrigeración

potVentilacionRef = 1.23 * 1 * float(c[7]) * float(c[8]) / 3.6 * (float(c[2]) - 23.9)

----- CARGA TOTAL -----

```
#potenciaCalefaccion = potEnvolvente + potVentiInfil
potCalefaccion = envolventeCal + potVentilacionInfiltracionCal

#sumatorio de la potRef de los cerramientos
pot=0
envolventeRef=[]
for i in range(24):
    for j in range(len(self.cogerCerramientos ())):
        pot+=envolvente[j][i]

    envolventeRef.append (pot)
    pot=0

#sumatorio de la potRef de las ventanas
pot=0
ventanasRef=[]
for i in range(24):
    for j in range(len(self.cogerVentanas())):
        pot+=ventanas[j][i]

    ventanasRef.append(pot)
    pot=0

#sumatorio de la potRef por horas
potencia=0
Refrigeracion=[]
for i in range(24):
    potencia = potOcupacionPorHoras[i] + potIluminacionPorHoras[i] + potEquiposPorHoras[i] +
    envolventeRef[i] + ventanasRef[i]
    Refrigeracion.append (potencia)
    potencia=0

#de todas ellas elijo la maxima (pico)
maxRefrigeracion=max(Refrigeracion)

#para saber la hora del día en que se produce la potencia pico de refrigeración
for i in range(len(Refrigeracion)):
    if maxRefrigeracion==Refrigeracion[i]:

#a esta le tengo que sumar la ventilación y la carga latente por ocupación
potRefrigeracion = maxRefrigeracion + potVentilacionRef + potRefLatente
```

```
#en potRefrigeracionMeses se añade la pot de Refrigeracion maxima de julio, agosto y septiembre
potRefrigeracionMeses.append (potRefrigeracion)

#elijo la máxima total (pot pico), entre la máxima de julio, max de agosto y max de septiembre
potRefrigeracionMesesMax = max(potRefrigeracionMeses)

#Mensaje en pantalla:
#se divide entre 1000 para pasar de W a kW
wx.MessageBox ('Potencia Calor: '+str (round (potCalefaccion/1000,2))+ ' kW \n\nPotencia Frio:
,
+str(round (potRefrigeracionMesesMax/1000,2))+ ' kW ', 'Nonresidential Load Calculations')

#####

def cogerPT(self):
    PTparaASHRAE = []
    for i in self.parent.panelEnvolvente.puentesTermicos:
        PTparaASHRAE.append([i[0],i[3],i[4]])
    return PTparaASHRAE

def cogerCerramientos(self):                ##los cerramientos son arrays dentro de array
    cerramientosParaAshrae = []
    for i in self.parent.panelEnvolvente.cerramientos:
        cerramientosParaAshrae.append([i[0],i[1],i[2],i[3],i[4],i[5],i[-1]])

#nombre(a[i][0]),tipo(a[i][1]),A(a[i][2]),U(a[i][3]),kg/m2(a[i][4]),Orientacion(a[i][5]),contacto(a[i][-1])
    return cerramientosParaAshrae

def cogerVentanas(self):                    ##las ventanas son objetos dentro de un array
    ventanasParaAshrae=[]
    for j in self.parent.panelEnvolvente.ventanas:
        ventanasParaAshrae.append ([j.descripcion, j.tipo, j.superficie, j.Uvidrio, j.Gvidrio,
        j.Umarco, j.orientacion, j.porcMarco, j.tieneProteccionSolar, j.altura,
        j.absortividadValor, j.cerramientoAsociado])
    return ventanasParaAshrae

def datos(self):
    datosNecesarios=[]

##    provincia  c[0]
    provincia=self.parent.panelDatosGenerales.provinciaChoice.GetStringSelection()
    datosNecesarios.append (provincia)
```

```
## ciudad c[1]
ciudad=self.parent.panelDatosGenerales.localidadChoice.GetStringSelection()
datosNecesarios.append (ciudad)

## temperaturas
temperaturas=temperaturasDiseno(provincia,ciudad)
tRefrigeracion=temperaturas[0] # c [2]
tCalefaccion=temperaturas[1] # c [3]
Oscilacion=temperaturas [2] # c [4]
datosNecesarios.append (tRefrigeracion)
datosNecesarios.append (tCalefaccion)
datosNecesarios.append (Oscilacion)

## anoConstruccion c[5]
anoConstruccion=self.parent.panelDatosGenerales.anoConstruccionChoice.GetStringSelection()
datosNecesarios.append (anoConstruccion)

## perfil de uso c[6]
tipoEdificio=self.parent.panelDatosGenerales.tipoEdificioChoice.GetStringSelection()
datosNecesarios.append (tipoEdificio)

## superfUtil c[7]
superfUtil=self.parent.panelDatosGenerales.superficie.GetValue()
datosNecesarios.append (superfUtil)

## alturaPlanta c[8]
alturaPlanta=self.parent.panelDatosGenerales.alturaMediaLibre.GetValue()
datosNecesarios.append (alturaPlanta)

## masaParticiones c[9]
masaParticiones=self.parent.panelDatosGenerales.masaParticiones.GetStringSelection()
datosNecesarios.append (masaParticiones)

## geografia
geografía = temperaturasDiseno(provincia,ciudad)
latitud =geografia[3] #c[10]
longitud =geografia[4] #c[11]
temperaturaPorHorasJulio = geografia[5] #c[12]
temperaturaPorHorasAgosto = geografia[6] #c[13]
temperaturaPorHorasSeptie = geografia [7] #c[14]
datosNecesarios.append (latitud)
datosNecesarios.append (longitud)
datosNecesarios.append (temperaturaPorHorasJulio)
datosNecesarios.append (temperaturaPorHorasAgosto)
datosNecesarios.append (temperaturaPorHorasSeptie)
return datosNecesarios

def cogerDatosTerciario(self):
a = []
a.append (self.ocupacion.GetValue ()) #0 Ocupacion
a.append (self.potencia.GetValue ()) #1 Potencia de los equipos
a.append (self.horarioInicioChoice.GetStringSelection ()) #2 horario inicio

for i in self.parent.panelInstalaciones.iluminacion:
a.append (i [2]) #3 Potencia iluminacion
```

a.append (i [7])

#4 Actividad

b = perfilUso (self.parent.panelDatosGenerales.tipoEdificioChoice.GetStringSelection ())

a.append (b)

#5 perfil de uso----> horas de funcionamiento

return a

def OnHorarioInicioChoice(self, event):

e = self.cogerDatosTerciario ()

x = float (e [2]) #inicio

y = float (e [5]) #horas de funcionamiento

z = x + y #fin = inicio + horas de funcionamiento (no se puede editar)

if z>24: #si fin>24horas, le resto 24

z=z-24

z = int (z) #para que en TextCtrl de la ventana aparezca un entero y no float

z = str (z) #para pasarselo a un TextCtrl se debe convertir en string

self.horarioFin.SetValue (z)

5.1.4. TablasASHRAE.py:

```
# -*- coding: cp1252 -*-
```

```
import wx
import math
import sys
import os
```

```
###Funcion que interpola en el eje X y en el eje Y
```

```
#Se le debe pasar:
```

```
# x,y= valor x, valor y, que nos interesa
```

```
# enX,enY= valores que forman ambos ejes
```

```
# matriz= matriz de valores
```

```
def interpola(x,y,enX,enY,matriz):
```

```
    if x <= enX[0]:
```

```
        indiceX = 0
```

```
    elif x>=enX[len(enX)-1]:
```

```
        indiceX = len(enX)-2
```

```
    else:
```

```
        for i in range(len(enX)):
```

```
            if x <= enX[i+1] and x>enX[i]:
```

```
                indiceX = i
```

```
                break
```

```
    if y <= enY[0]:
```

```
        indiceY = 0
```

```
    elif y>=enY[len(enY)-1]:
```

```
        indiceY = len(enY)-2
```

```
    else:
```

```
        for i in range(len(enY)):
```

```
            if y <= enY[i+1] and y>enY[i]:
```

```
                indiceY = i
```

```
                break
```

```
    valor1 = matriz [indiceY] [indiceX]
```

```
    valor2 = matriz [indiceY+1] [indiceX]
```

```
    valor3 = matriz [indiceY] [indiceX+1]
```

```
    valor4 = matriz [indiceY+1] [indiceX+1]
```

```
    a1 = valor1+ (valor3-valor1)/ (enX [indiceX+1]-enX [indiceX]) * (x-enX [indiceX])
```

```
    a2 = valor2+ (valor4-valor2)/ (enX [indiceX+1]-enX [indiceX]) * (x-enX [indiceX])
```

```
    c = a1+ (a2-a1)/(enY[indiceY+1]-enY[indiceY]) * (y-enY[indiceY])
```

```
    return c
```

```
###Funcion que interpola solo en el eje X (el eje Y es fijo):
```

```
#Se le debe pasar:
```

```
# x= valor x, que nos interesa
```

```
# enX,enY= valores que forman ambos ejes
```

```
# matriz= matriz de valores
```

```
def interpolaSoloX(x,enX,enY,matriz):
```

```
    indiceY = enY
```

```
    if x <= enX[0]:
```

```
        indiceX = 0
```

```
    elif x>=enX[len(enX)-1]:
```

```
        indiceX = len(enX)-2
```

```
    else:
```

```
        for i in range(len(enX)):
```

```
            if x>enX[i] and x <= enX[i+1]:
```

```
                indiceX = i
```

```
                break
```

```
    valor1 = matriz [indiceY] [indiceX]
```

```
    valor3 = matriz [indiceY] [indiceX+1]
```

```
    a1 = valor1+ (valor3-valor1)* float(x-enX [indiceX])/float (enX [indiceX+1]-enX [indiceX])
```

```
    return a1
```

```
#####  
#####  TABLAS RESIDENCIAL #####
```

```
###Unit Leakage Area Aul (cm2/m2).....tabla 3
```

```
# ASHRAE pag (17.6)
```

```
# Se le dedebe pasar:
```

```
# y= la calidad de la construccion
```

```
def tabla3(y):
```

```
    if y=='Hermética': indiceY=0
```

```
    elif y=='Buena': indiceY=1
```

```
    elif y=='Normal': indiceY=2
```

```
    elif y=='Con fugas': indiceY=3
```

```
    else: indiceY=4
```

```
    matriz = [
```

```
        [0.7],
```

```
        [1.4],
```

```
        [2.8],
```

```
        [5.6],
```

```
        [10.4]]
```

```
Aul = matriz [indiceY] [0]
return Aul

###Unit Leakage Area Aul (cm2/m2).....tabla 3a
# Se ha adaptado de la siguiente forma:
# año construccion--> CTE    --> 1.4
#          --> NBE    --> 2.8
#          --> Anterior --> 5.6
def tabla3a(y):
    if y=='C.T.E.': indiceY=0
    elif y=='NBE-CT-79': indiceY=1
    elif y=='Anterior': indiceY=2

    matriz =[
        [1.4],

        [2.8],

        [5.6],

        ]

    Aul = matriz [indiceY] [0]
    return Aul

###Infiltration Driving Force IDF (L/s·cm2).....tabla 5
# ASHRAE pag (17.6)
# Se le debe pasar:
# x= la Temperatura de diseño refrigeracion
# y= la Altura de Planta
def tabla5(x,y):
    enX = [-10,0,10,30,35,40]    ##Temperatura diseño refrigeracion
    enY = [2.5,3,4,5]           ##Altura Planta
    matriz =[
        [0.077, 0.069, 0.060, 0.031, 0.035, 0.04],    ##Altura = 2.5m
        [0.083, 0.072, 0.061, 0.032, 0.038, 0.043],    ##Altura = 3.0 m
        [0.093, 0.079, 0.065, 0.034, 0.042, 0.049],    ##Altura = 4.0 m
        [0.100, 0.086, 0.069, 0.036, 0.046, 0.055]]    ##Altura = 5.0 m

    IDF = interpola(x, y, enX, enY, matriz)
    return IDF
```

```
###Opaque Surface Cooling Coefficients.....tabla 7
# ASHRAE pag (17.9)
# Para calcular CFopq
# Se pasa, el cerramiento, si esta en contacto con el aire o no,
# la absorptancia del techo y la orientacion
def tabla7(cerramiento,contacto,AbsTecho,Orientacion):
```

```
    if cerramiento=='Cubierta':
        if contacto=='aire':
            OFt=0.62
            OFb=14.3*AbsTecho - 4.5
            OFr=-0.19
        elif contacto=='terreno':
            OFt=0.0
            OFb=0.0
            OFr=0.0

    elif cerramiento=='Fachada':
        if contacto=='aire':
            if Orientacion in ['Este','SE','Sur','SO','Oeste']:
                OFt=1.0
                OFb=8.2
                OFr=-0.36
            else:
                OFt=1.0
                OFb=0.0
                OFr=-0.36
        if contacto=='terreno':
            OFt=0.0
            OFb=0.0
            OFr=0.0
        if contacto=='edificio':
            OFt=0.0
            OFb=0.0
            OFr=0.0

    elif cerramiento=='Suelo':
        if contacto=='terreno':
            OFt=1.0
            OFb=0.0
            OFr=-0.06
        elif contacto=='aire':
            OFt=1
            OFb=0.0
            OFr=-0.36

    elif cerramiento=='Partición Interior':
        if contacto=='horizontal inferior':
            OFt=0.33
            OFb=0.0
            OFr=-0.28
        elif contacto=='horizontal superior':
            OFt=1.0
            OFb=38.3*AbsTecho - 7.0
            OFr=-0.36
        elif contacto=='vertical':
            OFt=1
```

```

    OFb=0
    OFr=-0.36
return [OFt, OFb, OFr]

###Opaque Doors Cooling Coefficient.....tabla 7a
# La misma que la anterior pero transformada para las puertas opacas.
def tabla7a(cerramiento,porcMarco,orientacion,proteccion):

    if cerramiento=='Hueco':
        if porcMarco=='100':
            if orientacion in ['Este','SE','Sur','SO','Oeste']:
                if proteccion==False:
                    OFt=1.0
                    OFb=8.2
                    OFr=-0.36
                else:
                    OFt=1.0
                    OFb=0.0
                    OFr=-0.36
            else:
                OFt=1.0
                OFb=0.0
                OFr=-0.36

        return [OFt, OFb, OFr]

###Peak Irradiance (W/m2).....tabla 10
# ASHRAE pag (17.9)
# Se obtiene ED, Et, Ed
# Para calcular PXI = Peak Exterior Irradiance
# Se le pasa:
# x= la orientacion
# y= la latitud
def tabla10(y, x):
    enX = [25,30,35,40,45]

    if y=='Norte': enY=0
    elif y=='NE' or y=='NO': enY=3
    elif y=='Este' or y=='Oeste': enY=6
    elif y=='SE' or y=='SO': enY=9
    elif y=='Sur': enY=12
    else: enY=15    #Techo

    matriz =[
        [106, 92, 84, 81,85],    ## ED    ##Norte
        [115, 103, 93, 84,76],    ## Ed
        [221, 195, 177, 166,162],    ## Et

        [449, 437, 425, 412,399],    ## ED    ##NE/NO
        [169, 162, 156, 151,147],    ## Ed
        [618, 599, 581, 563,546],    ## Et
    ]

```

```
[543, 552, 558, 560, 559],    ## ED    ##E/O
[196, 193, 190, 189, 188],    ## Ed
[739, 745, 748, 749, 747],    ## Et

[328, 369, 405, 436, 463],    ## ED    ##SE/SO
[203, 203, 204, 205, 207],    ## Ed
[531, 572, 609, 641, 670],    ## Et

[60, 139, 214, 283, 348],    ## ED    ##Sur
[193, 196, 200, 204, 209],    ## Ed
[253, 335, 414, 487, 557],    ## Et

[840, 827, 806, 776, 738],    ## ED    ##Horizontal
[170, 170, 170, 170, 170],    ## Ed
[1010, 997, 976, 946, 908] ]  ## Et
```

ED=interpolaSoloX(x, enX, enY, matriz)

Ed=interpolaSoloX(x, enX, enY+1, matriz)

Et=interpolaSoloX(x, enX, enY+2, matriz)

return ED, Ed, Et

```
###Shade Line Factors SLF.....tabla 12
# ASHRAE pag (17.10)
# Para calcular Fshd = fraction shaded
# Se le pasa:
# y= la orientacion
# x= la latitud
def tabla12 (y, x):
    enX = [25, 30, 35, 40, 45]          #Latitud

    if y=='Norte': enY=0                #Orientacion
    elif y=='NE' or y=='NO': enY=1
    elif y=='Este' or y=='Oeste': enY=2
    elif y=='SE' or y=='SO': enY=3
    elif y=='Sur': enY=4

    matriz = [
        [2.1, 1.4, 1.5, 1.7, 1.0],     ##Norte
        [1.5, 1.6, 1.2, 1.3, 1.3],     ##NE/NO
        [1.2, 1.1, 1.1, 1.1, 1.0],     ##E/O
        [1.8, 2.0, 1.7, 1.5, 1.6],     ##SE/SO
        [14.0, 6.9, 4.7, 3.3, 2.7]]    ##Sur

    SLF = interpolaSoloX(x, enX, enY, matriz)
    return SLF
```

###Fenestration Solar Load Factors FFs.....tabla 13

ASHRAE pag (17.10)

Es un factor para obtener CFfen

Se le pasa:

x= tipo de vivienda

y= orientacion

def tabla13(x, y):

 if x=='Unifamiliar': enX=0

 else: enX=1

 if y=='Norte': enY=0

 elif y=='NE': enY=1

 elif y=='Este': enY=2

 elif y=='SE': enY=3

 elif y=='Sur': enY=4

 elif y=='SO': enY=5

 elif y=='Oeste': enY=6

 elif y=='NO': enY=7

 else: enY=8

matriz =[

 [0.44, 0.27], ##Norte

 [0.21, 0.43], ##NE

 [0.31, 0.56], ##E

 [0.37, 0.54], ##SE

 [0.47, 0.53], ##Sur

 [0.58, 0.61], ##SO

 [0.56, 0.65], ##O

 [0.46, 0.57], ##NO

 [0.58, 0.73]] ##Horizontal

FFs = matriz [enY] [enX]

return FFs

#####

TERCARIO

#Por ejemplo:

#si el perfil de uso es 'Intensidad Media-12h' y el horario de uso empieza a las 6a.m.,

#finalizará a las: 6 + 12 = 18 h

#para ello necesito obtener el '12' en este caso

#por tanto, el choice que indica el fin del horario de uso, no se podrá editar y el programa lo pondrá de forma automática

def perfilUso(a):

```
if a=='Intensidad Baja - 8h' or a=='Intensidad Media - 8h' or a=='Intensidad Alta - 8h':
    x = 8
elif a=='Intensidad Baja - 12h' or a=='Intensidad Media - 12h' or a=='Intensidad Alta - 12h':
    x = 12
elif a=='Intensidad Baja - 16h' or a=='Intensidad Media - 16h' or a=='Intensidad Alta - 16h':
    x = 16
else:    ## 24h
    x = 24
```

return x

#Para saber desde que hora (inicio) hasta que hora (fin) está ocupado el edificio (están en funcionamiento los equipos y la iluminación)

def horarioFuncionamiento(inicio, fin):

```
horario=[]
if fin>inicio:
    for i in range(1,25):
        if i==24:
            if fin ==24:
                horario.append (1)
            else:
                horario.append (0)
        else:
            if i <(inicio) or i>=(fin):
                horario.append (0)
            else:
                horario.append (1)
else:
    for i in range(1,25):
        if i<fin or i>=inicio:
            horario.append (1)
        else:
            horario.append (0)
return horario
```



```
#En función de la actividad que se realice (la elegida por el usuario en el panel instalaciones--
>iluminación), el calor (sensible/latente) liberado será diferente, así como el porcentaje convectivo
def actividad(activity):
    if activity=='Administrativo en general' or activity=='Salas de diagnóstico' \
    activity=='Pabellones de exposición y ferias' or activity=='Aulas y laboratorios' \ or
    activity=='Habitaciones de hospital' or activity=='Zonas comunes':
        sensibleHeat=75      # (W)
        percentConvect=0.42
        latentHeat=55
    elif activity=='Almacenes, archivos, salas técnicas y cocinas':
        sensibleHeat=100
        percentConvect=0.51
        latentHeat=180
    elif activity=='Espacios deportivos':
        sensibleHeat=210
        percentConvect=0.46
        latentHeat=315
    else:      #por si acaso se señala 'Otros' en iluminación, pongo una actividad con un calor sensible medio
        sensibleHeat=150
        percentConvect=0.46
        latentHeat=240
    return sensibleHeat, percentConvect, latentHeat
```

```
#Un tipo de Radiant Time Series:
#for direct transmitted solar heat gain
#se asume que la energía radiante es distribuida en el suelo y el mobiliario
def SolarRTS(peso):
    if peso=='Ligera':
        RTF=[0.45,0.20,0.11,0.07,0.05,0.03,0.02,0.02,0.01,0.01,0.01,0.01,0.01,0.01,0.0,0.0,0.0,0.0,0.0,0.0]
    if peso=='Media':
        RTF=[0.29,0.15,0.10,0.07,0.06,0.05,0.04,0.03,0.03,0.03,0.02,0.02,0.02,0.02,0.01,0.01,0.01,0.01,
        0.01, 0.01, 0.01, 0, 0, 0]
    if peso=='Pesada':
        RTF=[0.27,0.13,0.07,0.05,0.04,0.04,0.03,0.03,0.03,0.03,0.03,0.03,0.02,0.02,0.02,0.02,0.02,0.02,
        0.02, 0.02, 0.02, 0.01, 0.01]
    return RTF
```

```
#Conduction Time Series
#ASHRAE Tabla 16 (pag. 18.24/18.25)
#para los muros de fachada. Los valores que se obtienen son los Conduction Time Factors (CTF)
def wallCTS(masa,U):
    if masa<40:
        if U<0.35:
            CTF=[0.2,0.25,0.31,0.2,0.11,0.05,0.03,0.02,0.01,0,0,0,0,0,0,0,0,0,0,0,0] #columna9
        elif U<0.41:
            CTF=[0.07,0.44,0.32,0.12,0.04,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #6
        elif U<0.6:
            CTF=[0.19,0.59,0.18,0.03,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #4 (1,2 y 4)
        else:
            CTF=[0.11,0.5,0.26,0.09,0.03,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #8
    elif masa<100:
        if U<0.8:
            CTF=[0.07,0.44,0.32,0.12,0.04,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #6 (3,5,6 y 7)
        else:
            CTF=[0.01,0.11,0.21,0.2,0.15,0.1,0.07,0.05,0.03,0.02,0.02,0.01,0.01,0.01,0,0,0,0,0,0,0,0]#24
    elif masa<120:
        CTF=[0.04,0.13,0.16,0.14,0.11,0.09,0.07,0.06,0.04,0.03,0.03,0.02,0.02,0.02,0.01,0.01,
            0.01, 0.01, 0, 0, 0, 0,0] #21 (21 y 25)
    elif masa<140:
        CTF=[0.01,0.02,0.06,0.09,0.09,0.09,0.08,0.07,0.06,0.06,0.05,0.05,0.04,0.04,0.03,0.03,
            0.03, 0.02, 0.02, 0.02, 0.02, 0.01, 0.01, 0] #10
    elif masa<160:
        CTF=[0.01,0.1,0.2,0.18,0.14,0.1,0.07,0.05,0.04,0.03,0.02,0.02,0.01,0.01,0.01,0.01,0,0,
            0, 0, 0, 0, 0,0] #27 (27 y 28)
    elif masa<180:
        CTF=[0.01,0.01,0.02,0.05,0.07,0.09,0.09,0.08,0.08,0.07,0.06,0.06,0.05,0.04,0.04,0.03,
            0.03, 0.02, 0.02, 0.02, 0.02, 0.02, 0.01, 0.01] #26
    elif masa<215:
        CTF=[0.05,0.14,0.17,0.15,0.12,0.09,0.07,0.05,0.04,0.03,0.02,0.02,0.01,0.01,0.01,0.01,
            0.01, 0, 0, 0, 0, 0,0] #11 (11 y 12)
    elif masa<250:
```



```

elif U<0.235:
    CTF=[0.01,0.12,0.25,0.22,0.15,0.1,0.06,0.04,0.02,0.01,0.01,0.01,0,0,0,0,0,0,0,0,0,0] #6
elif U<0.24:
    CTF=[0.01,0.17,0.31,0.24,0.14,0.07,0.04,0.02,0,0,0,0,0,0,0,0,0,0,0,0,0] #4
else:
    CTF=[0.06,0.45,0.33,0.11,0.03,0.01,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0,0] #1 (y 10)
elif masa<50:
    CTF=[0,0.07,0.18,0.18,0.15,0.11,0.08,0.06,0.05,0.03,0.03,0.02,0.01,0.01,0.01,
        0.01, 0, 0, 0, 0, 0, 0, 0] #7
elif masa<100:
    if U<0.28:
        CTF=[0.01,0.17,0.34,0.25,0.13,0.06,0.03,0.01,0,0,0,0,0,0,0,0,0,0,0,0,0] #5
    elif U<0.32:
        CTF=[0,0.1,0.22,0.2,0.14,0.1,0.07,0.05,0.04,0.03,0.02,0.01,0.01,0.01,0,0,0,0,0,0,0,0] #13
    else:
        CTF=[0.01,0.03,0.08,0.1,0.1,0.09,0.08,0.07,0.06,0.05,0.05,0.04,0.04,0.03,0.03,
            0.03, 0.02, 0.02, 0.02, 0.02, 0.01, 0.01, 0.01, 0] #8
elif masa<200:
    CTF=[0.01,0.02,0.08,0.11,0.11,0.1,0.09,0.07,0.06,0.05,0.05,0.04,0.03,0.03,0.03,
        0.02, 0.02, 0.02, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01] #14
elif masa<300:
    CTF=[0.02,0.02,0.05,0.06,0.07,0.07,0.06,0.06,0.06,0.05,0.05,0.05,0.04,0.04,
        0.04, 0.04, 0.03, 0.03, 0.03, 0.03, 0.03, 0.03, 0.02, 0.02] #17 (y 15)
elif masa<400:
    if U<0.3:
        CTF=[0.01,0.02,0.06,0.08,0.08,0.08,0.07,0.07,0.06,0.05,0.05,0.05,0.04,0.04,
            0.03, 0.03, 0.03, 0.03, 0.02, 0.02, 0.02, 0.02, 0.02, 0.02] #19
    else:
        CTF=[0.02,0.02,0.05,0.06,0.07,0.07,0.06,0.06,0.06,0.05,0.05,0.05,0.04,0.04,
            0.04, 0.04, 0.03, 0.03, 0.03, 0.03, 0.03, 0.03, 0.02, 0.02] #17
else:
    CTF=[0.03,0.03,0.03,0.05,0.06,0.06,0.06,0.06,0.06,0.05,0.05,0.05,0.05,0.04,0.04,
        0.04, 0.04, 0.04, 0.03, 0.03, 0.03, 0.03, 0.02, 0.02] #18
return CTF

```

#Sirve para vincular una orientación posible en CEX, con un número (necesario para realizar el método de cálculo)

#ASHRAE Capítulo 14, Tabla 4 (pag. 14.10)

```

def orientacion(x):
    if x=='Norte': ang=180
    elif x=='NE': ang=-135
    elif x=='Este': ang=-90
    elif x=='SE': ang=-45
    elif x=='Sur': ang=0
    elif x=='SO': ang=45
    elif x=='Oeste': ang=90
    elif x=='NO': ang=135
    return ang

```

```
#Beam Solar Heat Gain Coefficient
#para calcular Beam Solar Heat Gain (W)
#en función del ángulo Tita
def beamshgc(x):
    enX = [0, 40, 50, 60, 70,80]    ##ángulo de incidencia (Tita)
    enY=0
    matriz = [[0.7, 0.67, 0.64, 0.58, 0.45, 0.23]]    ##beam SHGC para type 5b

    b = interpolaSoloX(x, enX, enY, matriz)
    return b
```

```
def temperaturasDiseno (provincia, localidad):
    provincias = hola ()
    for i in range (len(provincias)):
        if provincia == provincias[i][0]:
            for j in range (len (provincias[i][1])):
                if localidad in provincias[i] [1][j][0]:
                    return [provincias[i][1][j][1],provincias[i][1][j][2],provincias[i][1][j][3],provincias[i][1][j][4],
                        provincias[i][1][j][5],provincias[i][2],provincias[i][3],provincias[i][4]]
                    ##TempDiseñoRefrigeracion persen: 5% ##TempDiseñoCalefaccion persen: 97.5%
##OscilacionMediaDiaria ##Latitud ##Longitud ##Temperatura (por horas) de Julio
##Temperatura (por horas) de Agosto ##Temperatura (por horas) de Septiembre
def hola ():
    provincias = [[u'A Coruña',
        [[u'Arteixo',26.0, 1.4, 14.5, 43.3,-8.5],
        [u'Carballo',26.0, 1.4, 14.5, 43.13,-6.5],
        [u'A Coruña',26.0, 1.4, 14.5, 43.22,-8.4],
        [u'Ferrol',26.0, 1.4, 14.5, 43.29,-8.23],
        [u'Narón',26.0, 1.4, 14.5, 42.8,-7.2],
        [u'Oleiros',26.0, 1.4, 14.5, 42.6,-9.0],
        [u'Ribeira',26.0, 1.4, 14.5, 42.34,-7.9],
        [u'Santiago de Compostela',28.4, 1.2, 17.5, 42.52,-7.97]],
        [17.1,16.8,15.6,14.8,15.3,15.5,16.5,17.7,19.5,21.0,23.2,25.1,    #31Julio
        26.4,27.5,28.2,26.7,26.0,24.6,22.7,21.3,19.8,18.8,18.1,17.3],[
        16.4,16.0,15.6,15.1,14.4,14.6,14.8,16.3,17.4,19.5,21.5,22.7,    #23Agosto
        24.3,24.5,25.3,24.7,24.0,22.8,21.1,19.5,18.3,17.3,16.9,16.5],[
        17.6,18.3,18.5,18.6,18.6,18.6,19.0,19.3,20.6,21.7,22.9,24.1,    #9Sept
        25.3,25.8,26.6,25.9,25.1,23.9,23.0,22.5,21.5,20.9,20.4,20.0]],
        [u'Álava',
        [[u'Amurrio',30.0,-2.4, 20.0, 43.03,-3.0],
        [u'Llodio',30.0,-2.4, 20.0, 43.15,-2.97],
        [u'Salvaterra',30.0,-2.4, 20.0, 42.85,-2.38],
        [u'Vitoria-Gasteiz',30.0,-2.4, 20.0, 42.51,-2.41]], [
        15.4,15.6,15.0,14.1,14.5,15.1,16.1,18.2,19.9,22.9,25.1,27.2,    #16Julio
        28.5,30.0,29.7,29.3,28.2,26.2,24.0,22.1,19.9,18.6,18.2,17.1],[
        15.4,14.7,14.3,13.9,13.2,13.5,14.7,16.1,18.6,20.8,23.7,26.3,    #14Agosto
        28.3,29.4,30.0,28.7,27.2,25.9,23.1,21.0,19.1,18.0,17.0,16.4],[
        13.9,14.3,13.8,13.3,13.5,13.4,13.8,14.7,16.2,19.1,21.3,23.5,    #13Sept
        24.7,26.1,26.6,26.1,24.6,22.5,20.7,19.4,18.0,17.0,16.4,15.6]],
        [u'Albacete',
```

- [[u'Albacete', 34.2,-3.0, 18.8, 39.0,-1.52],
 [u'Almansa', 34.2,-3.0, 18.8, 38.52,-1.06],
 [u'Hellin', 36.7, 1.5, 20.3, 38.31,-1.42],
 [u'Villarrobledo', 34.2,-3.0, 18.8, 39.16,-2.36]], [21.0,19.1,17.4,15.0,15.0,15.3,16.9,19.8,22.7,26.0,29.6,33.3, #25Julio
 35.7,37.2,37.8,37.1,34.7,32.1,28.8,25.9,23.5,21.7,20.4,19.1],[19.5,18.9,17.9,17.3,16.7,17.2,18.3,20.1,22.5,26.1,29.5,32.6, #18Agosto
 35.2,36.3,36.5,36.0,34.0,31.6,28.4,25.9,24.2,22.7,21.5,20.0],[18.5,18.6,17.9,17.4,17.1,17.2,17.7,19.5,21.7,24.4,27.4,30.2, #1Sept
 32.3,33.3,33.5,33.2,31.5,29.4,27.0,24.4,23.1,21.9,21.1,20.2]],
- [u'Alicante',
 [[u'Alcoy', 31.3, 0.6, 12.5, 38.42,-0.28],
 [u'Alicante', 31.3, 4.1, 12.5, 38.2,-0.29],
 [u'Benidorm', 31.3, 4.1, 12.5, 38.32,-0.08],
 [u'Crevillent', 31.3, 4.1, 12.5, 38.26,-0.8],
 [U'Dénia', 31.3, 4.1, 12.5, 38.5,-0.07],
 [u'Elche', 31.3, 4.1, 12.5, 38.15,-0.42],
 [u'Elda', 31.3, 1.6, 12.5, 38.29,-0.47],
 [u'Ibi', 31.3,-0.4, 12.5, 38.6,-0.57],
 [U'Jávea', 31.3, 4.1, 12.5, 38.78, 0.17],
 [u'Novelda', 31.3, 4.1, 12.5, 38.23,-0.45],
 [u'Orihuela', 31.3, 4.1, 12.5, 38.05,-0.57],
 [u'Petrer', 31.3, 1.6, 12.5, 38.48,-0.77],
 [u'San Vicente del Raspeig', 31.3, 4.1, 12.5, 38.4,-0.52],
 [u'Torrevieja', 31.3, 4.1, 12.5, 37.59,-0.4],
 [u'Villajoyosa', 31.3, 4.1, 12.5, 38.3,-0.14],
 [u'Villena', 31.3, 1.6, 12.5, 38.39,-0.52]], [23.4,22.4,21.8,21.4,21.1,21.4,22.2,23.7,25.5,28.3,30.7,33.1, #19Julio
 34.1,35.2,35.5,35.0,33.7,31.9,29.9,27.9,26.5,25.2,24.9,23.5],[23.2,22.7,22.0,21.1,20.8,20.7,21.3,22.9,24.8,27.5,30.2,32.6, #29Agosto
 34.3,35.4,35.7,35.1,33.4,31.8,29.5,27.6,26.2,24.9,23.9,23.5],[21.2,20.8,20.1,19.8,19.6,19.8,20.3,21.5,23.2,25.5,27.8,30.6, #13Sept
 32.9,33.7,34.0,33.7,31.8,30.0,27.8,26.3,25.2,23.9,23.1,22.1]],
- [u'Almería',
 [[u'Adra', 32.4, 4.3, 11.7, 36.45,-3.0],
 [u'Almería', 32.4, 4.3, 11.7, 36.5,-2.28],
 [u'El Ejido', 32.4, 4.3, 11.7, 36.7,-2.82],
 [u'Roquetas de mar', 32.4, 5.3, 11.7, 36.46,-2.37]], [22.1,20.9,20.3,19.1,18.8,19.2,20.0,21.3,23.9,26.3,28.6,31.1, #23Julio
 33.1,33.7,34.4,33.8,32.3,30.4,28.3,26.5,24.8,23.5,22.3,21.9],[23.3,23.4,23.2,22.8,22.9,22.8,23.5,24.8,26.2,28.6,30.8,32.4, #17Agosto
 34.2,34.8,35.0,34.4,33.4,32.1,30.1,28.2,27.1,26.2,25.3,25.2],[21.4,20.7,19.9,19.7,19.1,19.2,19.6,20.7,22.3,24.4,26.6,28.9, #20Sept
 30.5,31.7,31.6,30.5,29.9,27.8,26.2,24.9,23.9,23.0,22.4,21.8]],
- [u'Asturias',
 [[u'Avilés', 25.8, 1.2, 13.6, 43.33,-5.56],
 [u'Castrillón', 25.8, 1.2, 13.6, 43.38,-6.78],
 [u'Gijón', 25.8, 1.2, 13.6, 43.32,-5.42],
 [u'Langreo', 25.8, 1.2, 13.6, 43.18,-5.41],
 [u'Mieres', 25.8, 1.2, 13.6, 43.15,-5.46],
 [u'Oviedo', 25.8, 1.2, 13.6, 43.22,-5.5],
 [u'San Martín del Rey Aurelio', 25.8, 1.2, 13.6, 43.27,-5.62],
 [u'Siero', 25.8, 1.2, 13.6, 43.38,-5.67]], [19.3,19.1,18.6,17.5,18.1,18.0,19.0,19.9,21.1,22.9,24.1,25.9, #31Julio
 26.9,27.6,27.9,27.3,26.3,24.9,23.7,22.5,21.7,20.4,19.8,19.5],[16.1,15.8,14.7,13.9,13.0,12.7,14.2,15.3,17.3,19.2,21.1,23.0, #13Agosto
 25.0,25.9,25.1,24.7,24.2,22.2,20.8,18.9,17.8,16.4,15.8,15.5],[

- 15.8,15.2,14.0,13.8,12.7,12.7,13.3,14.8,16.6,18.5,21.2,23.7, #2Sept
25.4,26.8,26.9,26.2,24.7,23.0,21.0,19.2,18.1,16.6,15.9,15.5]],
- [u'Ávila',
[[u'Arenas de San Pedro',30.8,-4.5, 16.8, 40.12,-5.05],
[u'Arévalo',30.8,-4.5, 16.8, 41.04,-4.43],
[u'Ávila',30.8,-4.5, 16.8, 40.39,-4.42],
[u'El Barraco',30.8,-4.5, 16.8, 40.48,-4.63],
[u'Las Navas del Marqués',30.8,-4.5, 16.8, 40.6,-4.33]],
17.5,16.0,15.1,13.8,13.7,14.6,15.3,17.1,19.6,22.2,25.1,27.4, #28Julio
29.8,30.9,31.2,30.1,29.1,27.0,24.4,21.9,20.5,19.0,17.8,16.8],[
16.0,15.6,14.5,13.5,12.6,12.9,13.8,15.9,18.5,21.8,25.1,27.9, #15Agosto
29.9,31.6,31.5,31.1,29.2,26.9,24.0,22.0,19.7,18.2,17.2,15.9],[
12.1,11.1,11.1,10.1,9.0,9.3,10.3,11.4,13.9,17.9,20.8,24.5, #11Sept
26.9,28.3,28.9,28.1,25.9,23.6,20.4,18.2,16.3,15.1,13.7,12.9]],
- [u'Badajoz',
[[u'Almendralejo',36.8, 0.3, 19.7, 38.41,-6.24],
[u'Badajoz',36.8, 0.3, 19.7, 38.53,-6.58],
[u'Don Benito',36.8, 0.3, 19.7, 38.57,-5.52],
[u'Mérida',36.8, 0.3, 19.7, 38.55,-6.2],
[u'Villanueva de la Serena',36.8, 0.3, 19.7, 38.59,-5.48]], [
18.1,16.8,15.2,14.0,13.7,14.5,16.2,18.7,22.1,26.0,30.1,33.6, #20Julio
36.3,38.3,38.7,37.7,35.7,32.5,29.2,25.8,22.7,21.1,19.5,18.0],[
23.3,22.4,22.0,21.0,19.9,20.7,21.9,23.2,25.3,28.9,31.9,34.9, #16Agosto
37.3,38.3,38.8,38.3,36.2,34.3,31.3,28.9,27.5,25.8,24.2,23.7],[
17.1,16.3,16.1,14.9,14.4,14.2,15.4,17.0,19.9,23.8,27.6,30.9, #8Sept
33.5,35.5,35.8,35.0,32.7,29.9,26.7,24.2,22.0,20.4,19.4,18.1]],
- [u'Barcelona',
[[u'Badalona',30.0, 2.7, 9.2, 41.27, 2.15],
[u'Barberá del Vallés',30.0, 2.7, 9.2, 41.4, 2.01],
[u'Barcelona',30.0, 2.7, 9.2, 41.23, 2.11],
[u'Castelldefels',30.0, 2.7, 9.2, 41.28, 1.97],
[u'Cerdanyola del Valles',30.0, 2.7, 9.2, 41.3, 2.09],
[u'Cornellá de Llobregat',30.0, 2.7, 9.2, 41.42, 2.08],
[u'Gavá',30.0, 2.7, 9.2, 41.18, 2.0],
[u'Granollers',30.0, 2.7, 9.2, 41.37, 2.18],
[u'L'Hospitalet de Llobregat',30.0, 2.7, 9.2, 41.22, 2.08],
[u'Igualada',30.0, 2.7, 9.2, 41.35, 1.37],
[u'Manresa',30.0, 2.7, 9.2, 41.43, 1.5],
[u'El Masnou',30.0, 2.7, 9.2, 41.48, 2.32],
[u'Mataró',30.0, 2.7, 9.2, 41.32, 2.27],
[u'Mollet del Vallés',30.0, 2.7, 9.2, 41.33, 2.13],
[u'Moncada i Reixach',30.0, 2.7, 9.2, 41.48, 2.18],
[u'El Prat de Llobregat',30.0, 2.7, 9.2, 41.18, 2.03],
[u'Premia de Mar',30.0, 2.7, 9.2, 41.48, 2.47],
[u'Ripoll',30.0, 2.7, 9.2, 41.5, 2.17],
[u'Rubí',30.0, 2.7, 9.2, 41.48, 2.03],
[u'Sabadell',30.0, 2.7, 9.2, 41.33, 2.07],
[u'Sant Adrià de Besòs',30.0, 2.7, 9.2, 41.43, 2.23],
[u'Sant Boi de Llobregat',30.0,2.7,9.2,41.2,2.03],
[u'Sant Cugat del Vallés',30.0,2.7,9.2,41.47,2.08],
[u'Sant Feliú de Llobregat',30.0,2.7,9.2,41.23,2.03],
[u'Sant Joan Despí', 30.0, 2.7, 9.2, 41.37, 2.07],
[u'Sant Pere de Ribes',30.0,2.7,9.2,41.27,1.77],
[u'Sant Vicenç dels Horts',30.0,2.7,9.2,41.4,2.02],
[u'Santa Coloma de Gramanet',30.0,2.7,9.2,41.27,2.13],
[u'Terrasa',30.0, 2.7, 9.2, 41.34, 2.01],
[u'Vic',30.0, 2.7, 9.2, 41.56, 2.15],

- [u'Viladecans',30.0, 2.7, 9.2, 41.32, 2.0],
 [u'Vilafranca del Penedés',30.0,2.7,9.2,41.22,1.41],
 [u'Vilanova i la Geltrú',30.0,2.7,9.2,41.13,1.43]],
 20.3,20.0,20.1,19.5,19.3,19.9,20.9,22.0,23.4,25.7,27.7,29.7, #24Julio
 31.2,31.7,32.5,31.3,30.2,28.9,27.0,25.8,24.0,23.3,22.3,21.8],[
 19.7,19.3,18.5,18.3,17.9,17.9,18.8,20.2,22.5,24.6,27.2,29.2, #3Agosto
 30.7,31.8,31.8,31.1,30.2,28.5,26.6,24.7,23.0,22.0,21.3,19.9],[
 21.8,21.6,21.1,20.1,20.0,20.1,20.3,21.3,22.5,24.1,26.1,27.3, #5Sept
 28.6,29.6,30.2,29.5,28.5,26.9,25.3,24.7,23.4,22.7,22.5,21.8]],
 [u'Burgos',
 [[u'Aranda de Duero',31.4,-4.2, 21.5, 41.39,-3.42],
 [u'Burgos',31.4,-4.2, 21.5, 42.2,-3.42],
 [u'Miranda de Ebro',31.4,-4.2, 21.5, 42.41,-2.56]], [
 14.0,12.9,11.2,10.2,9.6,10.2,12.2,14.2,18.0,22.1,25.5,29.1, #30Julio
 32.1,33.5,34.5,33.1,31.2,28.2,24.6,21.4,18.7,17.0,15.6,13.8],[
 13.9,13.0,12.9,12.0,11.8,12.2,13.0,14.8,17.0,19.5,23.1,25.8, #30Agosto
 28.0,29.2,29.9,28.9,27.7,24.9,22.1,20.3,18.4,16.9,16.1,15.1],[
 11.4,10.3,9.3,8.5,7.8,8.3,8.5,10.4,13.5,16.5,19.9,23.0, #6Sept
 25.3,27.2,27.3,26.7,24.5,22.5,19.3,17.0,14.7,13.5,12.4,11.3]],
 [u'Cáceres',
 [[u'Cáceres',36.4, 1.2, 16.6, 39.28,-6.22],
 [u'Plasencia',36.4, 1.2, 16.6, 40.02,-6.06]], [
 20.5,19.5,18.5,17.5,17.5,18.3,19.4,22.0,24.8,28.2,31.6,34.9, #16Julio
 37.0,38.3,38.8,37.6,35.9,34.0,30.9,27.8,25.7,23.7,22.3,21.2],[
 22.7,22.4,22.4,22.2,22.3,22.3,22.9,24.7,26.6,28.6,31.1,33.2, #7Agosto
 35.2,35.8,35.9,35.4,34.2,32.4,30.4,28.9,27.2,26.2,25.1,24.5],[
 22.2,21.7,20.8,20.4,20.0,20.1,20.8,22.4,24.4,26.9,29.3,31.9, #4Sept
 34.3,35.7,35.6,35.2,33.8,31.2,29.4,27.3,25.6,24.7,23.5,23.1]],
 [u'Cádiz',
 [[u'Algeciras',35.7, 2.9, 18.6, 36.11,-5.27],
 [u'Arcos de la Frontera',35.7,2.9,18.6,36.45,-5.49],
 [u'Barbate',35.7, 2.9, 18.6, 36.19,-5.92],
 [u'Cádiz',35.7, 2.9, 18.6, 36.32,-6.18],
 [u'Chiclana de la Frontera',35.7,2.9,18.6,36.25,-6.09],
 [u'Jerez de la Frontera',35.7,2.9,18.6,36.41,-6.09],
 [u'La Línea de la Concepción',35.7,2.9,18.6,36.17,-5.35],
 [u'El Puerto de Santa María',35.7,2.9,18.6,36.36,-6.13],
 [u'Puerto Real',35.7,2.9,18.6,36.53,-6.18],
 [u'Rota',35.7, 2.9, 18.6, 36.6,-6.35],
 [u'San Fernando',35.7,2.9,18.6,36.28,-6.11],
 [u'San Roque',35.7,2.9,18.6,36.13,-5.23],
 [u'Sanlúcar de Barrameda',35.7,2.9,18.6,36.46,-6.21]], [
 23.9,23.5,23.5,22.9,22.9,23.0,23.9,24.3,25.0,26.7,27.6,29.4, #24Julio
 30.0,30.5,30.4,30.1,29.8,28.7,27.9,26.9,25.8,25.1,24.9,24.3],[
 22.4,22.1,22.0,21.0,20.4,20.6,21.3,22.3,23.7,26.0,27.6,29.3, #24Agosto
 30.6,31.8,31.7,30.8,30.2,29.1,27.0,25.6,24.6,24.1,23.2,22.7],[
 23.0,22.7,23.1,22.8,22.8,22.8,22.9,23.7,25.1,26.5,27.6,29.7, #16Sept
 30.3,31.3,31.4,31.0,30.1,29.1,27.8,26.9,25.8,24.9,24.8,24.5]],
 [u'Cantabria',
 [[u'Camargo',25.6, 2.6, 12.3, 43.4,-3.9],
 [u'Santander',25.6, 2.6, 12.3, 43.28,-3.48],
 [u'Torrelavega',25.6, 2.6, 12.3, 43.2,-4.02]], [
 18.4,18.4,18.3,17.8,17.8,18.4,19.2,19.9,21.3,22.1,23.2,24.3, #19Julio
 24.8,25.5,25.3,25.1,24.8,23.8,22.7,21.7,20.9,20.2,19.8,19.3],[
 18.8,18.8,19.3,19.4,19.4,19.6,19.9,20.6,21.5,22.7,23.6,24.6, #28Agosto
 25.3,26.2,26.5,25.9,25.2,24.5,23.4,22.5,22.1,21.4,21.1,20.7],[
 17.4,17.4,17.8,18.0,17.8,17.7,18.2,19.0,20.6,21.9,23.7,25.8, #5Sept

- 26.5,27.2,27.4,27.2,25.9,25.0,23.3,22.5,21.1,20.6,20.0,19.5]],
- [u'Castellón',
- [[u'Burriana',31.4, 4.4, 11.4, 39.53,-0.05],
- [u'Castellón de la Plana',31.4,4.4,11.4,39.59,-0.02],
- [u'La Vall d'Uixo',31.4,4.4,11.4,39.49,-0.09],
- [u'Vila-Real',31.4,4.4,11.4,39.56,-0.06],
- [u'Vinaroz',31.4, 4.4, 11.4, 40.28, 0.29]], [
- 22.0,21.6,20.7,20.8,20.7,21.5,22.4,23.3,25.2,27.1,29.0,30.4, #18Julio
- 31.7,32.6,32.9,32.3,31.7,30.2,28.0,26.5,25.4,24.6,23.4,22.9],[
- 24.0,23.3,22.6,22.1,21.1,21.6,22.4,23.3,25.4,27.5,29.5,31.3, #13Agosto
- 33.3,34.6,34.2,33.6,32.8,31.3,29.3,27.5,25.9,25.0,24.0,23.7],[
- 20.9,20.9,20.8,21.3,21.0,21.6,21.9,22.7,24.1,26.0,27.9,29.4, #9Sept
- 30.5,31.4,31.8,31.4,30.2,28.9,27.4,26.2,24.9,24.5,24.3,23.5]],
- [u'Ceuta',
- [[u'Ceuta',30.2, 8.4, 9.8, 35.9,-5.31]], [
- 16.3,16.1,16.1,16.2,16.1,16.1,17.1,18.7,20.7,22.7,24.9,27.0, #16Julio
- 28.5,29.7,29.9,29.4,28.1,26.2,24.1,22.6,21.2,19.8,19.0,18.2],[
- 20.0,19.6,19.6,19.6,19.6,19.7,20.5,21.2,23.5,25.4,27.6,29.4, #10Agosto
- 30.8,32.4,32.5,32.2,30.8,29.1,26.8,25.5,24.3,22.8,22.5,21.6],[
- 18.3,18.0,17.5,17.3,16.3,15.9,16.1,17.3,19.5,21.8,24.6,27.0, #27Sept
- 28.9,30.3,30.7,30.1,28.4,26.2,24.2,22.8,21.0,20.1,19.1,18.7]],
- [u'Ciudad Real',
- [[u'Alcázar de San Juan',36.2,-1.3,17.8,39.4,-3.12],
- [u'Ciudad Real',36.2,-1.3,17.8,39.0,-3.55],
- [u'Puertollano',36.2,-1.3, 17.8, 38.68,-4.07],
- [u'Tomelloso',36.2,-1.3, 17.8, 39.17,-3.02],
- [u'Valdepeñas',36.2,-1.3, 17.8, 38.77,-3.25]], [
- 21.8,20.5,20.1,18.8,18.9,19.5,20.7,22.6,25.2,28.2,31.8,34.6, #24Julio
- 36.7,38.0,38.0,37.4,36.0,33.3,30.4,28.6,26.0,24.8,23.3,22.4],[
- 17.9,17.6,16.8,16.5,15.9,16.4,17.6,19.9,22.7,26.3,30.2,33.5, #4Agosto
- 35.6,37.5,37.7,36.9,35.2,32.4,29.0,26.2,23.7,22.4,21.1,19.6],[
- 18.3,17.3,16.4,15.8,14.9,15.2,15.9,18.0,20.9,24.5,28.1,31.8, #6Sept
- 34.7,36.7,36.6,36.0,33.7,31.0,27.5,25.2,22.5,21.2,20.0,19.0]],
- [u'Córdoba',
- [[u'Baena',38.2, 1.4, 20.2, 37.37,-4.2],
- [u'Cabra',38.2, 1.4, 20.2, 37.28,-4.26],
- [u'Córdoba',38.2, 1.4, 20.2, 37.53,-4.47],
- [u'Lucena',38.2, 1.4, 20.2, 37.24,-4.29],
- [u'Montilla',38.2, 1.4, 20.2, 37.36,-4.38],
- [u'Priego de Córdoba',38.2,1.4,20.2,37.26,-4.12],
- [u'Puente Genil',38.2,1.4,20.2,37.23,-4.46]],[
- 23.8,22.2,21.6,20.1,20.4,20.8,21.8,24.1,27.2,30.2,34.1,37.3, #24Julio
- 39.6,41.2,41.2,40.6,39.0,36.4,33.1,30.3,28.0,26.3,25.3,24.0],[
- 21.5,20.5,18.9,17.3,16.0,16.5,17.7,19.8,23.4,27.5,32.4,35.9, #28Agosto
- 39.7,41.1,42.0,40.9,38.6,34.9,30.9,28.4,25.4,23.6,21.8,20.9],[
- 20.1,19.9,19.8,19.5,19.5,20.2,21.9,24.0,27.4,31.0,33.4, #1Sept
- 36.2,37.4,37.9,37.0,35.1,32.5,30.4,28.0,25.8,24.6,23.5,22.4]],
- [u'Cuenca',
- [[u'Cuenca',33.1,-3.1, 16.9, 40.04,-2.08],
- [u'Tarancón',33.1,-3.1, 16.9, 40.01,-3.01]], [
- 17.6,17.0,16.8,16.0,16.0,16.7,18.0,20.0,22.3,25.3,28.6,31.0, #21Julio
- 33.4,34.6,35.3,34.1,32.9,30.0,27.6,24.9,23.3,21.2,19.9,19.4],[
- 18.1,17.5,16.6,15.8,15.6,16.1,16.8,18.8,21.5,24.6,27.8,30.4, #10Agosto
- 32.9,34.1,34.4,33.7,32.0,29.5,26.7,24.2,22.8,21.2,19.7,18.6],[
- 15.3,13.9,13.0,11.6,10.8,10.4,11.8,13.5,16.5,19.7,23.8,27.3, #4Sept
- 29.7,31.6,31.7,31.0,28.7,26.3,23.0,20.3,18.7,16.9,15.8,14.5]],
- [u'Girona',

- [[u'Blanes',32.2,-1.9, 17.2, 41.4, 2.48],
 [u'Figueres',32.2,-1.9, 17.2, 42.16, 2.58],
 [u'Girona',32.2,-1.9, 17.2, 41.59, 2.49],
 [u'Olot',32.2,-1.9, 17.2, 42.11, 2.3],
 [u'Salt',32.2,-1.9, 17.2, 41.98]], [19.3,19.7,19.2,18.9,19.1,19.2,20.8,22.1,24.4,26.6,29.3,31.7, #26Julio
 33.7,34.5,34.6,34.2,32.7,31.0,28.6,26.4,24.7,23.2,22.4,21.9],[20.1,19.9,19.3,18.7,18.4,18.6,19.2,20.5,22.6,25.4,27.8,30.2, #31Agosto
 31.9,33.0,33.3,32.7,31.3,29.3,27.1,25.4,24.1,22.6,21.6,21.2],[16.0,15.5,14.9,15.1,14.1,14.4,15.0,16.4,18.8,21.0,23.9,26.4, #7Sept
 28.2,29.6,29.9,28.9,27.8,25.5,23.2,21.6,19.9,18.6,18.0,16.7]],
- [u'Granada',
 [[u'Almuñecar',36.0,-2.4, 21.0, 36.44,-3.41],
 [u'Baza',36.0,-2.4, 21.0, 37.3,-2.46],
 [u'Granada',36.0,-2.4, 21.0, 37.11,-3.35],
 [u'Guadix',36.0,-2.4, 21.0, 37.18,-3.09],
 [u'Loja',36.0,-2.4, 21.0, 37.1,-4.1],
 [u'Motril',36.0,-2.4, 21.0, 36.44,-3.31]], [18.0,16.6,15.1,13.2,13.3,14.5,15.8,18.3,21.5,25.4,29.5,33.4, #15Julio
 36.0,37.4,38.0,36.8,35.0,31.8,28.4,24.9,23.0,20.8,18.8,17.5],[20.2,19.2,18.3,16.5,15.3,15.8,17.3,19.4,22.9,26.4,30.2,33.3, #3Agosto
 36.3,37.9,38.3,37.4,35.1,32.3,29.3,26.1,24.5,22.2,20.6,19.3],[20.4,19.4,18.0,17.5,15.7,16.1,16.8,18.5,21.2,24.4,28.6,31.5, #4Sept
 34.8,35.8,36.3,35.0,33.0,30.5,27.7,25.1,22.9,21.8,20.5,19.3]],
- [u'Guadalajara',
 [[u'Azuqueca de Henares',31.8,-3.7,22.3,40.57,-3.27],
 [u'Guadalajara',31.8,-3.7, 22.3, 40.38,-3.1],
 [u'Molina de Aragón',31.8,-6.7,22.3,40.51,-1.88]], [18.7,16.9,14.6,12.9,13.0,13.2,15.4,17.5,20.8,24.5,28.2,31.2, #21Julio
 34.1,35.7,36.5,35.6,33.2,30.2,27.1,24.1,20.8,19.8,18.0,16.9],[17.7,16.9,16.0,15.0,14.2,13.9,15.4,17.6,20.1,23.3,27.0,29.9, #15Agosto
 32.2,33.9,34.1,33.4,31.9,29.1,25.9,23.8,21.6,19.5,18.4,17.4],[15.2,14.4,13.0,12.3,11.4,11.5,12.5,14.5,17.2,21.0,24.8,28.4, #6Sept
 31.0,32.8,33.3,32.6,30.1,27.1,23.9,21.6,19.5,17.7,16.5,15.7]],
- [u'Guipúzcoa',
 [[u'Éibar',28.6, 1.0, 14.7, 43.11,-2.28],
 [u'Irún',28.6, 1.0, 14.7, 43.2,-1.47],
 [u'Mondragón',28.6, 1.0, 14.7, 43.05,-2.3],
 [u'Rentería',28.6, 1.0, 14.7, 43.17,-1.58],
 [u'San Sebastián',28.6,1.0,14.70,43.19,-1.59],
 [u'Zarautz',28.6, 1.0, 14.7, 43.17,-2.1]], [18.9,19.5,19.2,19.5,19.9,19.5,20.1,21.0,22.6,23.5,24.7,25.3, #31Julio
 26.4,27.3,27.3,27.2,26.4,24.7,24.1,22.9,22.2,21.9,21.1,20.7],[18.6,18.1,18.1,17.7,18.4,18.4,18.6,19.4,20.6,21.9,23.1,24.2, #3Agosto
 25.3,25.8,25.6,25.3,24.6,24.0,22.7,21.8,20.9,19.9,19.7,19.5],[18.9,18.6,18.7,18.8,18.7,18.8,19.4,19.7,20.6,21.7,22.9,24.4, #4Sept
 25.0,25.7,25.7,25.4,24.8,24.0,22.8,21.9,21.3,20.7,20.3,20.3]],
- [u'Huelva',
 [[u'Almonte',34.4, 3.6, 16.9, 37.26,-6.52],
 [u'Ayamonte',34.4, 3.6, 16.9, 37.13,-7.24],
 [u'Huelva',34.4, 3.6, 16.9, 37.16,-6.57],
 [u'Isla Cristina',34.4,3.6,16.9,37.22,-7.32],
 [u'Lepe',34.4, 3.6, 16.9, 37.25,-7.2]], [20.7,20.4,19.8,19.3,18.6,18.9,20.3,22.1,24.9,27.7,30.8,33.6, #31Julio
 35.7,37.1,37.5,36.8,35.0,32.6,29.6,27.9,25.3,24.1,23.0,22.1],[20.8,20.0,19.6,18.1,17.5,17.6,18.8,20.4,22.8,26.0,29.3,31.9, #15Agosto
 33.7,35.5,36.3,35.2,33.2,31.2,28.3,26.1,24.3,22.9,21.9,20.7],[

- 20.9,20.0,19.5,19.2,18.6,18.3,19.2,20.6,23.0,25.7,28.1,31.0, #1Sept
33.2,34.1,34.8,34.0,32.4,30.3,27.9,26.1,24.4,23.2,22.6,21.5]],
- [u'Huesca',
[[u'Barbastro',33.4,-2.5, 16.1, 42.02, 0.08],
[u'Fraga',33.4,-2.5, 16.1, 41.32, 0.21],
[u'Huesca',33.4,-2.5, 16.1, 42.08,-0.24],
[u'Jaca',33.4,-2.5, 16.1, 42.34,-0.33],
[u'Monzón',33.4,-2.5, 16.1, 41.55, 0.11],
[u'Sabiñanigo',33.4,-2.5, 16.1, 42.31,-0.22]], [
- 20.1,19.1,18.1,17.3,17.1,17.9,18.8,20.7,23.4,25.8,28.5,31.3, #24Julio
32.8,34.3,34.7,34.1,32.3,30.2,27.9,25.6,23.7,21.9,21.0,20.4],[
- 20.4,18.7,17.5,16.4,15.7,16.5,18.0,19.8,23.2,26.2,29.1,32.2, #4Agosto
34.6,36.0,36.4,35.6,34.0,31.5,29.0,26.0,23.9,22.3,20.8,19.7],[
- 18.2,17.5,17.6,17.7,17.2,17.2,17.6,18.9,20.8,22.5,25.4,27.7, #7Sept
29.5,30.1,30.7,30.2,29.0,27.2,25.2,23.1,22.4,21.1,20.0,19.4]],
- [u'Illes Balears',
[[u'Calviá',32.2, 1.5, 15.6, 39.57, 2.52],
[u'Ciudadella de Menorca',32.2,1.5,15.6,40.0,3.75],
[u'Eivissa',32.2, 1.5, 15.6, 38.54, 1.26],
[u'Inca',32.2, 1.5, 15.6, 39.43, 2.9],
[u'Llucmajor',32.2, 1.5, 15.6, 39.29, 2.9],
[u'Mahón',32.2, 1.5, 15.6, 39.53, 4.25],
[u'Manacor',32.2, 1.5, 15.6, 38.34, 3.2],
[u'Palma de Mallorca',32.2,1.5,15.6,39.35,2.65],
[u'Santa Eulalia del Río',32.2,1.5,15.6,38.98,1.52]],[
- 23.4,23.0,22.7,22.1,22.5,22.5,23.3,24.5,25.8,27.3,29.2,31.5, #31Julio
32.2,33.0,33.3,33.1,32.0,30.8,29.1,27.7,26.1,25.5,24.8,24.1],[
- 24.6,23.7,23.3,22.6,22.0,21.9,22.6,23.5,25.0,26.4,28.1,29.9, #29Agosto
31.1,31.7,32.3,31.6,30.6,29.4,28.2,26.4,25.9,25.4,24.6,24.0],[
- 23.6,23.4,23.1,22.7,22.4,22.5,23.2,24.0,25.6,27.0,29.2,30.6, #1Sept
32.5,33.3,33.4,32.7,31.9,30.3,28.9,27.7,26.7,26.1,25.4,24.7]],
- [u'Jaén',
[[u'Alcalá la Real',35.0,2.6,13.0,37.28,-3.56],
[u'Andújar',35.0, 2.6, 13.0, 38.02,-4.03],
[u'Jaén',35.0, 2.6, 13.0, 37.46,-3.47],
[u'Linares',35.0, 2.6, 13.0, 38.06,-3.38],
[u'Martos',35.0, 2.6, 13.0, 37.44,-3.58],
[u'Úbeda',35.0, 2.6, 13.0, 38.01,-3.23]], [
- 21.3,20.3,19.5,18.4,18.6,18.7,20.4,22.4,25.5,28.6,31.8,34.7, #19Julio
37.0,38.4,38.4,38.0,36.1,33.7,30.8,27.8,26.0,24.2,23.0,21.9],[
- 24.9,24.2,23.6,22.1,21.5,22.4,22.8,25.4,27.2,29.8,32.9,35.9, #11Agosto
38.1,39.1,39.6,38.7,37.2,34.7,32.4,30.1,28.2,27.0,26.1,25.2],[
- 21.0,20.1,19.5,18.4,17.6,17.7,18.6,20.4,22.7,25.3,28.7,31.3, #1Sept
33.7,35.1,35.5,34.3,33.1,30.2,28.0,25.7,24.3,23.0,21.8,20.8]],
- [u'La Rioja',
[[u'Alfaro',33.2,-1.1, 19.2, 42.1,-1.45],
[u'Arnedo',33.2,-1.1, 19.2, 42.14,-2.05],
[u'Calahorra',33.2,-1.1, 19.2, 42.18,-1.59],
[u'Haro',33.2,-1.1, 19.2, 42.36,-2.48],
[u'Logroño',33.2,-1.1, 19.2, 42.28,-2.27],
[u'Santo Domingo de la Calzada',33.2,-1.1, 19.2, 42.26,-2.57]], [
- 19.9,18.6,17.9,16.7,16.8,17.6,18.3,20.3,22.9,25.7,28.4,31.4, #16Julio
33.0,34.3,34.3,33.9,32.5,30.3,27.8,25.5,23.5,21.9,20.7,19.6],[
- 18.7,18.2,17.8,17.4,17.1,17.3,18.3,19.7,22.1,24.9,27.5,30.1, #24Agosto
31.6,33.2,33.7,32.8,31.1,29.3,27.0,24.5,23.0,22.0,20.9,20.1],[
- 17.0,16.3,15.5,14.5,14.0,14.0,14.8,16.9,18.9,21.8,25.2,27.5, #2Sept
30.0,31.2,31.7,30.8,29.3,26.8,24.3,22.6,20.2,19.4,18.3,17.2]],

[u'Las Palmas',

[[u'Arrecife',29.4, 13.5, 10.5, 28.95,-13.53],
[u'Arucas',29.4, 13.5, 10.5, 28.07,-15.31],
[u'Galdar',29.4, 13.5, 10.5, 28.15,-15.65],
[u'Ingenio',29.4, 13.5, 10.5, 27.92,-15.43],
[u'Las Palmas de Gran Canaria',29.4,13.5,10.5,28.06,-15.25],
[u'San Bartolomé de Tirajana',29.4,13.5,10.5,28.56,-15.35],
[u'Santa Lucía',29.4,13.5,10.5,28.72,-17.73],
[u'Telde',29.4, 13.5, 10.5, 27.58,-15.25]], [
22.0,22.1,22.5,22.1,22.5,22.6,22.9,23.9,25.4,26.9,28.3,29.7, #27Julio
30.8,31.7,31.8,31.0,30.7,29.4,27.6,26.5,25.6,25.1,24.6,24.3],[
23.4,22.8,23.1,22.9,22.5,22.6,23.0,23.9,25.1,25.7,27.3,27.7, #11Agosto
29.3,29.3,29.6,29.5,28.9,27.8,26.8,26.0,25.4,24.7,24.7,23.8],[
23.0,22.7,22.2,22.2,22.0,21.9,22.5,23.3,24.5,26.1,28.0,29.0, #1Sept
30.5,31.3,31.3,31.3,30.3,28.8,27.6,26.0,25.3,24.8,24.6,23.7]],

[u'León',

[[u'León',30.0,-3.8, 16.9, 42.36,-5.34],
[u'Ponferrada',30.0,-3.8, 16.9, 42.33,-6.35],
[u'San Andrés del Rabanedo',30.0,-3.8,16.9,42.62,-5.6]], [
15.7,14.6,13.2,11.8,12.0,12.4,14.2,16.7,19.4,22.5,26.5,29.5, #24Julio
31.8,33.5,33.6,32.6,31.1,28.3,25.3,22.5,20.0,18.0,16.9,15.5],[
15.5,15.3,14.5,13.5,13.7,13.8,15.4,17.4,19.8,23.2,26.3,29.0, #9Agosto
31.1,32.2,32.7,32.2,30.2,27.8,25.5,22.8,20.8,19.1,18.1,17.1],[
11.5,10.6,9.9,9.0,8.6,8.0,8.3,9.7,11.7,15.2,19.4,22.8, #30Sept
25.6,27.5,27.8,26.8,25.1,21.6,19.1,17.0,15.4,13.9,12.7,11.8]],

[u'Lugo',

[[u'Lugo',28.0,-2.5, 20.3, 43.01,-7.33],
[u'Monforte de Lemos',28.0,-2.5, 20.3, 42.31,-7.3],
[u'Viveiro',28.0,-2.5, 20.3, 43.39,-7.38]], [
16.6,16.3,15.7,15.5,15.4,16.2,16.9,18.5,19.4,21.9,24.3,25.7, #20Julio
27.4,28.6,28.7,28.1,27.2,25.8,23.9,21.8,20.5,19.2,18.2,17.8],[
15.2,14.6,13.7,12.5,11.6,12.2,13.3,15.4,17.4,20.6,23.1,25.7, #19Agosto
27.8,29.2,29.6,29.0,27.5,25.0,22.4,20.5,18.5,16.9,15.8,15.0],[
12.7,11.9,11.6,11.0,10.8,11.0,11.1,12.6,14.7,17.7,20.7,23.4, #19Sept
25.8,26.9,27.7,26.7,25.0,22.5,20.1,18.3,17.0,15.6,14.8,14.2]],

[u'Lleida',

[[u'Balaguer', 34.0,-2.8, 17.2, 41.47, 0.48],
[u'Lérida', 34.0,-2.8, 17.2, 41.37, 0.38],
[u'La Seu d'Urgell',34.0,-2.8, 17.2, 42.22, 1.28]], [
24.5,24.2,23.8,23.6,23.5,23.9,24.7,26.5,28.1,29.9,32.2,34.3, #19Julio
35.6,36.2,36.4,36.0,35.3,33.2,32.0,30.0,28.1,27.5,26.4,25.7],[
20.3,19.0,18.2,16.7,17.3,17.5,18.5,20.3,22.7,25.0,28.2,30.9, #5Agosto
32.7,34.1,34.4,33.9,32.0,29.9,27.2,24.8,23.3,21.9,20.7,19.8],[
20.4,20.0,20.0,19.4,19.3,19.5,19.9,21.0,22.6,24.7,26.9,29.2, #8Sept
30.7,32.2,31.9,31.6,30.0,28.5,26.9,25.0,23.7,23.2,22.1,21.6]],

[u'Madrid',

[[u'Alcalá de Henares',35.2,-2.4, 18.7, 40.28,-3.22],
[u'Alcobendas',35.2,-2.4, 18.7, 40.32,-3.38],
[u'Alcorcón',35.2,-2.4, 18.7, 40.2,-3.5],
[u'Aranjuez',35.2,-2.4, 18.7, 40.01,-3.38],
[u'Arganda de Rey',35.2,-2.4, 18.7, 40.19,-3.42],
[u'Colmenar Viejo',35.2,-2.4, 18.7, 40.39,-3.47],
[u'Collado Villalba',35.2,-2.4, 18.7, 40.36,-3.59],
[u'Coslada',35.2,-2.4, 18.7, 40.43,-3.57],
[u'Fuenlabrada',35.2,-2.4, 18.7, 40.17,-3.48],
[u'Getafe',35.2,-2.4, 18.7, 40.18,-3.44],
[u'Leganés',35.2,-2.4, 18.7, 40.19,-3.46],

- [u'Madrid',35.2,-2.4, 18.7, 40.24,-3.41],
 [u'Majadahonda',35.2,-2.4, 18.7, 40.47,-3.87],
 [u'Móstoles',35.2,-2.4, 18.7, 40.2,-3.52],
 [u'Parla',35.2,-2.4, 18.7, 40.23,-3.77],
 [u'Pinto',35.2,-2.4, 18.7, 40.14,-3.43],
 [u'Pozuelo de Alarcón',35.2,-2.4, 18.7, 40.43,-3.82],
 [u'Rivas-Vaciamadrid',35.2,-2.4, 18.7, 40.33,-3.52],
 [u'Las Rozas de Madrid',35.2,-2.4, 18.7, 40.48,-3.87],
 [u'San Fernando de Henares',35.2,-2.4, 18.7, 40.43,-3.53],
 [u'San Sebastián de los Reyes',35.2,-2.4, 18.7, 40.55,-3.63],
 [u'Torrejón de Ardoz',35.2,-2.4, 18.7, 40.27,-3.29],
 [u'Tres Cantos',35.2,-2.4, 18.7, 40.36,-3.43],
 [u'Valdemoro',35.2,-2.4, 18.7, 40.2,-3.67]], [
 21.3,20.6,20.2,19.9,19.8,20.1,21.8,23.1,25.2,27.8,30.5,32.9, #4Julio
 34.3,35.4,36.0,35.3,33.7,31.6,29.8,27.9,26.3,24.4,23.8,22.6],[
 22.2,21.0,20.9,20.4,20.3,20.8,21.2,23.2,25.3,27.8,30.2,32.4, #7Agosto
 34.3,35.4,35.5,35.0,33.7,31.6,29.3,27.8,26.1,24.9,23.5,23.1],[
 17.9,17.2,16.6,15.8,15.8,15.2,16.2,17.9,20.2,22.8,25.1,27.6, #4Sept
 29.9,31.9,31.3,31.1,29.4,27.4,25.0,22.8,21.5,20.2,19.5,18.4]],
 [u'Málaga',
 [[u'Antequera',33.2, 5.8, 14.7, 37.01,-4.33],
 [u'Benalmádena',33.2, 5.8, 14.7, 36.6,-4.57],
 [u'Estepona',33.2, 5.8, 14.7, 36.25,-5.09],
 [u'Fuengirola',33.2, 5.8, 14.7, 36.32,-4.38],
 [u'Málaga',33.2, 5.8, 14.7, 36.43,-4.25],
 [u'Marbella',33.2, 5.8, 14.7, 36.3,-4.54],
 [u'Mijas',33.2, 5.8, 14.7, 36.6,-4.63],
 [u'Rincón de la Victoria',33.2, 5.8, 14.7, 36.72,-4.3],
 [u'Ronda',33.2, 5.8, 14.7, 36.45,-5.17],
 [u'Torremolinos',33.2, 5.8, 14.7, 36.38,-4.3],
 [u'Vélez-Málaga',33.2, 5.8, 14.7, 36.47,-4.05]], [
 21.3,20.4,19.8,19.4,19.0,19.7,20.6,22.2,24.1,26.2,29.0,31.0, #18Julio
 32.5,33.8,34.0,33.6,32.3,30.4,28.2,26.3,24.6,24.0,22.8,21.6],[
 25.3,23.7,23.5,23.0,22.0,22.2,22.6,24.2,25.6,27.4,29.2,31.2, #6Agosto
 32.4,33.0,33.4,32.6,31.9,30.7,28.5,27.3,26.3,25.1,24.6,23.9],[
 24.2,24.0,23.3,23.7,22.8,22.6,23.3,24.2,25.4,26.7,28.7,29.3, #3Sept
 30.9,31.7,31.8,31.5,30.5,29.2,28.2,27.3,26.1,25.4,24.9,24.2]],
 [u'Melilla',
 [[u'Melilla', 30.2, 8.4, 9.8, 35.32,-2.95]], [
 23.7,22.6,22.6,22.1,21.9,22.1,22.8,23.8,25.1,26.8,28.7,30.1, #28Julio
 30.9,32.1,32.3,31.7,30.9,29.3,27.6,26.9,25.6,24.7,24.0,23.7],[
 23.1,22.7,22.2,21.7,21.1,21.1,22.2,23.2,24.7,26.5,28.2,30.3, #10Agosto
 31.9,32.5,32.6,32.3,31.7,29.5,28.2,26.5,25.3,24.5,23.7,23.6],[
 21.2,20.7,19.6,19.6,19.1,19.3,19.6,21.1,22.4,24.2,26.2,28.0, #20Sept
 29.6,31.3,31.1,30.4,29.4,27.5,26.0,24.9,23.6,22.2,22.0,21.6]],
 [u'Murcia',
 [[u'Águilas',34.8, 3.5, 16.1, 37.24,-1.35],
 [u'Alcantarilla',34.8, 3.5, 16.1, 37.97,-1.22],
 [u'Caravaca de la Cruz',34.8, 3.5, 16.1, 38.08,-1.52],
 [u'Cartagena',34.8, 3.5, 16.1, 37.36,-0.59],
 [u'Cieza',34.8, 3.5, 16.1, 38.15,-1.25],
 [u'Jumilla',34.8, 3.5, 16.1, 38.29,-1.2],
 [u'Lorca',34.8, 3.5, 16.1, 37.41,-1.42],
 [u'Molina de Segura',34.8, 3.5, 16.1, 38.03,-1.13],
 [u'Murcia',34.8, 3.5, 16.1, 37.59,-1.07],
 [u'Torre-Pacheco',34.8, 3.5, 16.1, 37.74,-0.95],
 [u'Totana',34.8, 3.5, 16.1, 37.46,-1.3],

- [u'Yecla',34.8, 3.5, 16.1, 38.37,-1.07]], [
 21.7,21.0,20.6,19.7,19.7,19.6,21.0,22.0,23.8,25.9,28.2,30.5, #30Julio
 31.7,32.6,32.8,32.1,31.3,29.3,27.7,25.5,24.4,23.5,22.7,22.1],[
 22.8,21.5,21.1,20.3,19.5,19.6,20.0,21.5,23.6,25.5,27.7,29.7, #24Agosto
 31.3,32.8,32.9,32.4,31.2,29.2,27.7,25.9,24.4,23.7,22.8,22.0],[
 22.8,21.9,20.7,19.7,18.7,19.0,19.6,20.5,22.4,24.9,27.1,30.0, #12Sept
 31.2,33.0,32.9,32.5,31.0,28.9,26.9,25.4,24.0,23.0,21.7,21.5]],
- [u'Navarra',
 [[u'Alsasua',32.4,-2.0, 19.2, 42.54,-2.17],
 [u'Barañáin',32.4,-2.0, 19.2, 42.48,-1.7],
 [u'Vera de Bidasoa',32.4,-2.0, 19.2, 43.28,-1.68],
 [u'Olite',32.4,-2.0, 19.2, 42.3,-1.42],
 [u'Pamplona',32.4,-2.0, 19.2, 42.49,-1.38],
 [u'Sangüesa',32.4,-2.0, 19.2, 42.35,-1.17],
 [u'Tudela',32.4,-2.0, 19.2, 42.04,-1.36]], [
 12.2,11.6,10.6,10.0,10.1,10.6,12.4,15.2,18.0,21.4,25.1,27.9, #5Julio
 30.2,31.7,32.0,31.1,29.2,27.0,24.1,20.7,18.2,16.8,15.0,13.5],[
 17.9,16.9,15.6,14.6,14.6,14.8,16.6,18.2,20.2,22.9,26.1,28.7, #4Agosto
 30.5,32.0,32.4,31.4,30.2,28.2,25.5,23.1,21.3,19.8,19.3,17.8],[
 14.7,14.1,13.4,12.5,12.1,12.0,12.9,14.2,17.5,20.4,23.6,26.7, #4Sept
 28.7,29.9,30.7,30.0,28.1,25.7,22.8,20.6,18.9,17.5,16.5,15.4]],
- [u'Ourense',
 [[u'A Rúa',33.9,-1.4, 21.5, 42.4,-7.1],
 [u'O Barco de Valdeorras',33.9,-1.4, 21.5, 42.25,-6.59],
 [u'Ourense',33.9,-1.4, 21.5, 42.2,-7.52],
 [u'Verín',33.9,-1.4, 21.5, 41.57,-7.27]], [
 20.6,20.3,19.9,19.1,19.1,19.6,20.6,22.6,24.1,26.3,28.9,31.4, #27Julio
 32.9,34.1,34.1,33.5,32.2,30.6,28.2,26.8,25.0,23.7,22.5,21.8],[
 18.4,17.3,16.3,15.7,15.6,16.4,17.7,19.8,22.6,25.4,28.8,31.4, #2Agosto
 33.4,35.2,35.2,34.4,32.8,30.4,28.0,25.1,22.9,21.4,20.4,19.5],[
 15.1,14.5,13.7,13.0,11.8,12.2,12.9,14.5,17.5,20.7,24.1,26.6, #3Sept
 29.5,31.3,31.6,30.6,28.7,25.7,23.4,21.3,19.3,18.0,16.9,15.6]],
- [u'Palencia',
 [[u'Aguilar de Campoo',30.8,-3.6, 17.7, 42.48,-4.15],
 [u'Guardo',30.8,-3.6, 17.7, 42.78,-4.83],
 [u'Herrera de Pisuerga',30.8,-3.6, 17.7, 42.35,-4.2],
 [u'Palencia',30.8,-3.6, 17.7, 42.0,-4.32],
 [u'Saldaña',30.8,-3.6, 17.7, 42.32,-4.44]], [
 14.7,14.5,13.2,12.8,12.4,13.4,14.5,16.8,19.6,22.9,26.1,28.9, #8Julio
 31.2,32.2,32.2,31.5,29.7,27.9,25.0,22.2,19.4,18.3,17.2,15.9],[
 15.7,15.0,14.1,13.4,12.7,13.0,14.2,16.4,19.2,22.7,26.6,30.0, #18Agosto
 32.8,34.1,34.6,33.7,31.9,28.8,25.2,23.0,20.8,18.6,17.7,16.6],[
 16.9,16.4,16.2,15.5,15.2,15.6,15.7,16.9,18.7,21.0,23.4,25.5, #3Sept
 27.3,28.5,28.8,27.7,26.9,25.1,22.8,21.5,20.0,18.7,18.1,17.6]],
- [u'Pontevedra',
 [[u'Cangas',29.4, 3.3, 16.1, 42.26,-8.78],
 [u'A Estrada',29.4, 3.3, 16.1, 42.42,-8.29],
 [u'Lalín',29.4, 3.3, 16.1, 42.4,-8.07],
 [u'Marín',29.4, 3.3, 16.1, 42.38,-8.7],
 [u'Pontevedra',29.4, 3.3, 16.1, 42.26,-8.39],
 [u'Redondela',29.4, 3.3, 16.1, 42.17,-8.6],
 [u'Vigo',29.4, 3.3, 16.1, 42.15,-8.43],
 [u'Vilagarcía de Arousa',29.4, 3.3, 16.1, 42.36,-8.46]], [
 16.0,15.3,14.4,13.7,13.6,14.2,15.3,17.0,19.3,21.3,23.7,25.9, #28Julio
 27.7,28.7,28.9,28.1,26.8,25.3,23.1,20.9,19.5,17.8,16.8,16.6],[
 19.0,17.8,17.4,16.4,15.5,15.7,16.6,17.8,20.4,23.0,26.0,28.3, #27Agosto
 30.3,31.7,31.7,31.2,29.9,27.6,24.8,23.1,21.4,20.3,19.4,18.1],[

- 13.3,13.2,12.9,12.6,12.1,11.9,12.7,14.2,16.5,18.9,22.1,24.4, #17Sept
27.0,28.0,28.4,27.7,26.2,23.7,21.6,19.6,17.9,17.2,16.0,15.3]],
- [u'Salamanca',
[[u'Béjar',32.0,-4.4, 20.9, 40.23,-5.46],
[u'Ciudad Rodrigo',32.0,-4.4, 20.9,40.36,-6.32],
[u'Peñaranda de Bracamonte',32.0,-4.4, 20.9, 40.54,-5.13],
[u'Salamanca',32.0,-4.4, 20.9, 40.57,-5.4],
[u'Vitigudino',32.0,-4.4, 20.9, 41.01,-6.26]], [
- 16.3,15.6,15.1,14.6,14.2,15.3,16.4,18.8,22.0,24.8,27.7,30.9, #23Julio
33.2,34.5,34.5,34.2,32.2,29.6,27.0,24.6,22.2,20.8,19.4,18.2],[
- 17.3,16.0,15.6,14.5,14.6,14.9,16.2,18.0,20.8,23.6,26.7,29.1, #2Agosto
31.4,32.5,32.2,31.8,30.8,28.4,25.7,23.7,21.3,19.8,18.5,17.8],[
- 14.3,13.6,13.0,13.0,12.2,12.0,13.0,14.1,16.8,19.2,21.9,25.0, #10Sept
26.7,28.2,28.6,27.7,26.5,24.1,21.7,19.5,18.4,16.7,15.7,15.2]],
- [u'Santa Cruz de Tenerife',
[[u'Arona',30.0, 14.6, 9.5, 28.08,-16.67],
[u'Icod de los Vinos',30.0,14.6,9.5,28.35,-16.7],
[u'La Orotava',30.0,14.6,9.5,28.38,-16.52],
[u'Puerto de la Cruz',30.0,14.6,9.5,28.4,-16.55],
[u'Los Realejos',30.0,14.6,9.5,28.3,-16.3],
[u'San Cristóbal de la Laguna',30.0,14.6,9.5,28.48,-16.32],
[u'Santa Cruz de Tenerife',30.0,14.6,9.5,28.45,-16.23],
[u'Tacoronte',30.0, 14.6, 9.5, 28.47,-16.4]], [
- 25.9,24.6,23.8,23.1,22.1,22.3,23.1,24.0,25.7,27.4,29.6,31.4, #28Julio
32.8,33.8,34.1,33.6,32.5,30.8,29.5,27.4,26.3,25.7,25.1,24.2],[
- 22.1,22.2,21.8,21.2,21.0,20.8,21.9,22.9,24.4,26.8,29.0,30.8, #12Agosto
32.8,32.9,33.2,32.9,31.6,30.4,28.1,26.7,25.6,24.7,24.0,23.2],[
- 24.3,23.3,22.2,21.7,21.3,20.8,20.8,22.3,24.3,25.7,27.8,29.6, #5Sept
31.2,31.8,32.0,31.4,30.7,29.3,27.4,26.1,25.0,24.0,23.7,22.7]],
- [u'Segovia',
[[u'Cuéllar',32.1,-3.4, 16.0, 41.23,-4.21],
[u'El Espinar',32.1,-3.4, 16.0, 40.72,-4.25],
[u'San Ildefonso',32.1,-3.4, 16.0, 40.9,-4.0],
[u'Segovia',32.1,-3.4, 16.0, 40.57,-4.07]], [
- 16.0,14.9,13.7,12.7,12.7,13.9,15.3,17.7,21.4,24.2,28.0,30.9, #2Julio
33.9,35.5,35.4,34.6,33.3,29.9,27.0,24.0,21.3,19.7,17.9,17.0],[
- 16.3,15.1,14.0,13.2,13.6,14.0,15.4,17.7,20.5,24.2,27.7,30.7, #4Agosto
33.3,35.1,35.1,34.3,32.8,30.0,26.9,23.7,21.3,19.7,18.5,17.1],[
- 18.2,17.3,16.1,15.1,13.5,13.9,14.3,15.9,18.0,20.5,23.2,25.8, #10Sept
28.1,28.9,29.1,28.7,26.8,24.9,22.9,21.3,19.8,18.3,18.0,16.7]],
- [u'Sevilla',
[[u'Alcalá de Guadaira',37.6, 4.5, 17.4, 37.2,-5.5],
[u'Camas',37.6, 4.5, 17.4, 37.4,-6.03],
[u'Carmona',37.6, 4.5, 17.4, 37.28,-5.38],
[u'Coria del Río',37.6, 4.5, 17.4, 37.18,-6.04],
[u'Dos Hermanas',37.6, 4.5, 17.4, 37.17,-5.55],
[u'Écija',37.6, 4.5, 17.4, 37.32,-5.05],
[u'Lebrija',37.6, 4.5, 17.4, 36.55,-6.05],
[u'Mairena del Aljarafe',37.6, 4.5, 17.4, 37.33,-6.07],
[u'Morón de la Frontera',37.6, 4.5, 17.4, 37.07,-5.28],
[u'Los Palacios y Villafranca',37.6, 4.5, 17.4, 37.1,-5.55],
[u'La Rinconada',37.6, 4.5, 17.4, 37.48,-5.97],
[u'San Juan de Aznalfarache',37.6, 4.5, 17.4,37.37,-6.03],
[u'Sevilla',37.6, 4.5, 17.4, 37.23,-5.59],
[u'Utrera',37.6, 4.5, 17.4, 37.11,-5.46]], [
- 24.3,23.1,23.2,22.4,22.5,22.5,23.9,26.0,28.7,31.2,34.9,37.1, #21Julio
39.6,41.0,41.1,40.5,38.6,36.3,33.9,31.4,29.2,27.9,26.6,25.8],[

- 22.2,21.5,20.7,19.4,18.4,18.7,20.5,22.2,25.5,28.5,32.4,35.7, #9Agosto
 38.2,39.8,39.9,39.2,37.1,34.7,31.7,28.4,26.4,25.2,23.8,22.5],[
 22.1,21.2,20.7,20.5,19.5,19.5,20.4,21.8,23.7,26.6,29.4,32.8, #16Sept
 35.1,36.2,36.5,35.8,33.9,31.6,29.1,27.1,25.5,24.6,23.4,22.6]],
- [u'Soria',
 [[u'Ágreda',31.0,-4.8, 19.6, 41.49,-1.54],
 [u'Almazán',31.0,-4.8, 19.6, 41.29,-2.32],
 [u'Burgo de Osma',31.0,-4.8, 19.6, 41.35,-3.04],
 [u'San Leonardo de Yagüe',31.0,-4.8, 19.6,41.83,-3.07],
 [u'Soria',31.0,-4.8, 19.6, 41.46,-2.28]], [
 17.1,16.6,15.8,15.4,15.4,15.8,17.1,19.8,21.9,24.9,27.9,30.5, #23Julio
 32.7,34.1,34.3,33.4,32.2,29.6,27.1,24.6,22.6,21.0,19.4,18.5],[
 16.5,16.7,16.0,15.6,15.1,15.5,16.6,18.5,20.9,23.9,26.6,29.1, #12Agosto
 31.1,32.8,32.8,31.9,30.5,28.5,25.8,23.5,21.9,20.5,19.5,18.6],[
 14.0,13.5,13.1,12.3,11.5,11.8,12.6,14.0,16.6,19.6,22.4,25.6, #10Sept
 28.1,29.3,29.9,28.6,26.9,24.4,21.9,20.0,18.7,17.0,16.1,14.8]],
- [u'Tarragona',
 [[u'Reus', 30.8, 0.5, 13.2, 41.1, 1.06],
 [u'Tarragona', 30.8, 0.5, 13.2, 41.07, 1.25],
 [u'Tortosa',30.8, 0.5, 13.2, 40.49, 0.52],
 [u'Valls',30.8, 0.5, 13.2, 41.17, 1.15],
 [u'El Vendrell',30.8, 0.5, 13.2, 41.13, 1.32]], [
 22.1,21.1,20.0,19.3,19.1,19.9,20.6,22.7,24.2,26.4,28.5,31.2, #22Julio
 32.6,33.2,34.1,33.2,31.8,30.3,28.1,26.3,24.8,23.5,22.8,22.0],[
 22.9,22.3,22.6,22.0,21.7,21.9,22.9,24.1,26.2,28.2,31.0,33.0, #18Agosto
 34.9,36.0,36.4,35.6,34.3,32.8,30.4,28.4,27.1,26.2,25.1,24.3],[
 21.1,20.8,19.9,19.8,18.9,19.0,19.5,20.6,21.9,24.5,26.9,29.3, #16Sept
 31.5,32.1,32.7,32.2,30.5,29.2,26.6,25.2,23.9,23.3,22.2,21.5]],
- [u'Teruel',
 [[u'Alcañiz',32.6,-6.1, 21.5, 41.02,-0.08],
 [u'Andorra',32.6,-6.1, 21.5, 40.97,-0.43],
 [u'Calamocha',32.6,-6.1, 21.5, 40.55,-1.17],
 [u'Calanda',32.6,-6.1, 21.5, 40.56,-0.14],
 [u'Teruel',32.6,-6.1, 21.5, 40.2,-1.06]], [
 18.3,17.5,16.9,16.2,15.8,16.6,18.0,20.2,22.9,25.8,28.8,32.1, #16Julio
 33.8,35.5,35.4,34.5,33.0,30.7,28.1,25.5,23.3,21.8,20.3,19.6],[
 15.8,14.5,13.6,12.6,12.7,13.2,14.2,16.6,20.0,23.6,27.7,30.8, #5Agosto
 33.9,35.7,35.6,34.8,32.9,29.9,26.6,24.0,21.5,19.5,17.6,16.4],[
 16.2,16.0,14.9,13.9,13.0,12.8,13.8,15.8,18.1,21.3,25.2,28.6, #12Sept
 31.4,32.7,33.2,32.6,30.2,28.1,24.7,22.4,20.3,19.0,17.9,16.7]],
- [u'Toledo',
 [[u'Illescas',35.3,-3.6, 20.4, 40.08,-3.51],
 [u'Seseña',35.3,-3.6, 20.4, 40.1,-3.65],
 [u'Talavera de la Reina',35.3,-3.6, 20.4, 39.58,-4.5],
 [u'Toledo',35.3,-3.6, 20.4, 39.51,-4.01],
 [u'Torrijos',35.3,-3.6, 20.4, 39.59,-4.17]], [
 22.8,21.6,20.1,18.6,19.1,19.5,20.8,23.1,26.5,29.7,33.2,36.1, #23Julio
 38.4,40.0,40.3,39.7,37.4,35.1,32.1,29.0,27.2,25.0,23.9,22.6],[
 21.9,21.4,20.8,20.0,19.9,19.8,20.8,22.9,25.5,29.1,32.0,35.0, #17Agosto
 37.3,38.6,39.3,38.0,36.6,34.5,31.3,29.0,26.7,24.9,24.0,23.1],[
 17.7,17.5,17.4,17.1,16.5,16.6,17.8,19.3,21.7,24.5,27.7,30.7, #5Sept
 32.9,34.6,34.6,33.8,32.4,29.7,27.4,25.4,22.9,21.9,20.9,20.2]],
- [u'Valencia',
 [[u'Alaquàs',31.3, 5.5, 12.3, 39.45,-0.45],
 [u'Aldaia',31.3, 5.5, 12.3, 38.51,-0.31],
 [u'Algemesi',31.3, 5.5, 12.3, 39.12,-0.26],
 [u'Alzira',31.3, 5.5, 12.3, 39.09,-0.26],

[u'Burjassot',31.3, 5.5, 12.3, 39.3,-0.25],
[u'Carcaixent',31.3, 5.5, 12.3, 39.08,-0.28],
[u'Catarroja',31.3, 5.5, 12.3, 39.24,-0.24],
[u'Cullera',31.3, 5.5, 12.3, 39.1,-0.15],
[u'Gandía',31.3, 5.5, 12.3, 38.58,-0.11],
[u'Manises', 31.3, 5.5, 12.3, 39.29,-0.28],
[u'Mislata', 31.3, 5.5, 12.3, 39.47,-0.42],
[u'Oliva', 31.3, 5.5, 12.3, 38.55,-0.07],
[u'Ontinyent', 31.3, 5.5, 12.3, 38.5,-0.36],
[u'Paterna', 31.3, 5.5, 12.3, 39.3,-0.26],
[u'Quart de Poblet', 31.3, 5.5, 12.3, 39.48,-0.43],
[u'Sagunto', 31.3, 5.5, 12.3, 39.41,-0.16],
[u'Sueca', 31.3, 5.5, 12.3, 39.12,-0.2],
[u'Torrent', 31.3, 5.5, 12.3, 39.26,-0.28],
[u'Valencia', 31.3, 5.5, 12.3, 39.28,-0.22],
[u'Xàtiva', 31.3, 5.5, 12.3, 39.0,-0.31],
[u'Xirivella', 31.3, 5.5, 12.3, 39.47,-0.42]], [
21.7,20.5,19.7,18.9,19.0,19.5,20.3,21.8,23.8,26.2,28.2,30.2, #13Julio
31.6,32.7,33.5,32.3,31.1,29.6,27.2,26.0,24.1,22.9,22.1,21.1],[
23.5,23.4,22.4,21.8,21.1,21.7,22.0,23.7,25.3,27.1,28.9,30.8, #13Agosto
32.3,32.9,33.9,32.8,31.7,30.3,28.5,27.0,25.8,25.0,24.1,23.7],[
20.5,20.4,20.5,20.1,19.7,19.8,20.3,21.4,22.9,24.7,26.6,28.5, #3Sept
29.8,30.6,31.3,30.3,29.9,28.1,26.2,24.8,24.1,23.0,22.6,21.9]],
[u'Valladolid',
[[u'Arroyo de la Encomienda',33.2,-2.8, 19.1, 41.2,-4.6],
[u'Laguna de Duero',33.2,-2.8, 19.1, 41.58,-4.72],
[u'Medina del Campo',33.2,-2.8, 19.1, 41.18,-4.55],
[u'Tordesillas',33.2,-2.8, 19.1, 41.3,-5.0],
[u'Valladolid',33.2,-2.8, 19.1, 41.39,-4.44]], [
17.6,16.6,15.2,13.5,13.9,14.5,16.4,18.7,22.1,25.6,29.4,32.9, #17Julio
35.2,36.6,36.7,36.1,34.4,31.4,28.6,25.5,22.7,20.6,18.7,17.9],[
17.8,16.2,15.0,13.7,13.9,14.2,15.8,17.7,20.6,24.6,28.0,30.9, #3Agosto
33.7,35.3,35.3,34.7,32.9,29.8,26.8,24.2,22.2,20.1,18.6,17.8],[
15.2,15.2,14.8,14.1,14.0,14.2,15.0,16.5,18.4,21.2,24.3,27.1, #7Sept
29.0,30.0,30.7,30.1,28.7,25.8,23.9,22.2,20.2,18.5,18.2,17.0]],
[u'Vizcaya',
[[u'Barakaldo',28.8, 1.2, 16.3, 43.2,-2.57],
[u'Basauri',28.8, 1.2, 16.3, 43.02,-2.4],
[u'Bilbao',28.8, 1.2, 16.3, 43.15,-2.55],
[u'Durango',28.8, 1.2, 16.3, 43.13,-2.4],
[u'Erandio',28.8, 1.2, 16.3, 43.3,-2.95],
[u'Galdakao',28.8, 1.2, 16.3, 43.23,-2.83],
[u'Getxo',28.8, 1.2, 16.3, 43.37,-3.02],
[u'Leioa',28.8, 1.2, 16.3, 43.33,-2.98],
[u'Portugalete',28.8, 1.2, 16.3, 43.32,-3.02],
[u'Santurtzi',28.8, 1.2, 16.3, 43.33,-3.03],
[u'Sestao',28.8, 1.2, 16.3, 43.3,-3.0]], [
18.7,17.6,17.1,16.4,16.3,16.8,18.0,19.2,21.3,23.4,25.9,27.9, #27Julio
29.3,30.6,30.6,30.3,28.7,27.2,25.2,23.2,21.4,20.0,19.4,19.1],[
12.9,12.9,12.4,12.3,11.9,12.0,13.4,15.1,16.6,20.2,23.1,25.9, #28Agosto
28.2,29.4,30.0,29.3,27.4,25.0,22.3,20.5,18.7,17.0,16.0,15.3],[
17.7,17.4,17.0,16.4,16.0,16.0,16.1,17.4,19.0,21.9,24.7,27.2, #23Sept
28.8,30.1,30.4,29.8,28.9,26.2,24.4,22.7,21.2,20.1,19.1,19.1]],
[u'Zamora',
[[u'Benavente',33.0,-3.2, 18.5, 42.0,-5.41],
[u'Puebla de Sanabria',33.0,-3.2, 18.5, 42.05,-6.63],
[u'Toro',33.0,-3.2, 18.5, 41.31,-5.24],

```
[u'Villalpando',33.0,-3.2, 18.5, 41.52,-5.25],
[u'Zamora',33.0,-3.2, 18.5, 41.3,-5.45]], [
  18.3,17.8,17.9,18.0,18.1,17.9,19.8,21.4,23.6,26.7,29.4,32.6, #30Julio
  34.4,35.3,36.0,35.5,33.4,31.4,28.4,26.3,24.4,23.4,22.0,21.4],[
  19.0,18.4,18.2,17.9,17.3,17.3,19.0,20.4,22.5,25.1,27.7,30.9, #12Agosto
  32.7,33.3,34.1,33.5,31.3,29.6,27.6,25.1,23.6,22.2,21.2,20.4],[
  13.2,13.3,12.1,11.9,11.2,11.3,12.0,14.1,16.7,20.4,24.0,27.5, #11Sept
  30.6,32.2,32.7,31.8,29.7,26.5,23.8,20.8,18.9,17.3,16.1,15.4]],
[u'Zaragoza',
  [[u'Calatayud',34.5,-1.1, 17.1, 41.21,-1.38],
  [u'Ejea de los Caballeros',34.5,-1.1, 17.1, 42.07,-1.09],
  [u'Tarazona',34.5,-1.1, 17.1, 41.54,-1.43],
  [u'Utebo',34.5,-1.1, 17.1, 41.72,-1.0],
  [u'Zaragoza',34.5,-1.1, 17.1, 41.39,-0.52]], [
  21.2,20.1,19.9,19.2,18.9,19.5,20.8,22.9,25.8,28.5,31.5,33.9, #17Julio
  36.1,37.1,37.4,36.6,35.4,33.3,30.6,28.2,26.2,24.7,23.4,22.5],[
  19.8,19.1,18.8,18.4,18.3,18.8,19.8,21.8,24.3,26.8,29.7,32.8, #7Agosto
  34.9,35.9,36.3,35.9,34.4,31.6,29.3,27.0,25.2,23.9,22.6,21.4],[
  18.7,18.5,18.9,18.1,18.0,18.2,18.7,19.5,21.6,24.2,26.7,29.1, #9Sept
  31.0,32.3,32.4,31.9,30.5,28.6,26.2,24.4,23.0,22.1,21.0,20.5]]]
```

return provincias

6. ANÁLISIS DE LOS RESULTADOS

En este apartado se explican con detalle los ficheros expuestos en el apartado anterior. Cada línea de código es comentada, así como todas las clases y métodos, diferenciando los datos que provienen del programa, de los que aporta el usuario y los datos que son propios del método empleado (ASHRAE).

6.1. PluginsDES.py

En este fichero están incluidos todos los plugins que ofrece el programa. Salvo para el plugin ASHRAE, para el resto están desarrollados los pasos seguidos para su elaboración. Para este trabajo técnico, lo que interesa es lo siguiente: cuando se hace click sobre PluginsDes, la última opción es ASHRAE. Al hacer click sobre ASHRAE, el programa hace lo siguiente:

- Si estamos en residencial, CE3X irá a ASHRAE.py y ofrecerá una ventana
- Si estamos en Terciario, ya sea Pequeño Terciario o Gran Terciario, el programa irá a ASHRAETerciario.py y ofrecerá una ventana completamente diferente.

6.2. ASHRAE.py

Este fichero se ejecuta cuando se hace clic en ASHRAE, siempre y cuando se esté en el programa Residencial.

En teoría, los procesos para calefacción y refrigeración deben ser idénticos. Hay simplificaciones aceptables para calefacción mientras que en refrigeración, se utilizan diferentes aproximaciones.

Para el cálculo de la carga de refrigeración se ha utilizado el Residential Load Factor (RLF) method, que tiene las siguientes características:

- Para el cálculo en residencial, los sistemas deben ser diseñados para las condiciones representativas de carga máxima, no para condiciones extremas. Se asume una ocupación normal, no la máxima que puede darse de forma ocasional.
- Los ventiladores para ventilación que operen de forma intermitente se supondrá que están apagados.
- Condiciones de diseño: las condiciones interiores, propuestas por ASHRAE, y se fijan en 24°C y una humedad entre 50% y 60% para verano, y 20°C y humedad aproximada del 30% para invierno. Las condiciones exteriores, ASHRAE fija, en refrigeración, una temperatura de diseño de bulbo húmedo del 1% y coincidente con bulbo seco. Para calefacción, utiliza una temperatura de diseño de bulbo seco del 99%.

PROCEDIMIENTO

- 1.- Condiciones de diseño
 - 2.- Cantidades de los componentes: áreas y longitudes requeridas
 - 3.- Opaque Surface Factors
 - 4.- Window Factors
 - 5.- Cargas de la envolvente
 - 6.- Ventilación e infiltración
 - 7.- Ganancias internas
 - 8.- Pérdidas por distribución (no se han considerado en este Trabajo Técnico)
 - 9.- Carga sensible total
- La parte latente se evalúa de forma separada.

Para el cálculo de la carga de calefacción, se dan las siguientes suposiciones:

- No existen ganancias internas de calor
- No existen ganancias de calor solar
- No hay almacenamiento de calor
- Pérdidas de calor evaluadas instantáneamente

Por tanto, el cálculo se reduce a un cálculo básico de: $Q = U \cdot A \cdot dT$ en el que las transmitancias térmicas de los cerramientos y huecos se hacen imprescindibles. También tiene en cuenta los puentes térmicos y los cálculos de la ventilación e infiltración.

6.3. ASHRAETerciario.py

Este fichero se ejecuta cuando se hace clic en ASHRAE, cuando se está en Pequeño Terciario o en Gran Terciario.

Primero se van a explicar los cálculos de las cargas de refrigeración ya que presenta grandes diferencias con otros métodos existentes.

Modos de entrada de calor

- Radiación solar a través de las superficies transparentes
- Conducción de calor a través de muros y tejados exteriores
- Conducción de calor a través de techos, suelos y particiones interiores
- Calor generado en el espacio por ocupantes, iluminación y aparatos
- Ventilación e infiltración del aire exterior
- Otros

Calor sensible

Es el añadido directamente al espacio condicionado por conducción, convección y/o radiación.

Calor latente

La ganancia ocurre cuando la humedad es añadida al espacio. Para mantener una tasa de humedad constante, el vapor de agua debe condensar en un aparato de refrigeración y ser eliminado a la misma velocidad que es añadido al espacio.

Ganancia de calor radiante: la energía radiante debe ser primero absorbida por las superficies que engloban el espacio (muros, suelos, techos) y objetos en el espacio (mobiliario). Cuando estas superficies y objetos llegan a estar a una temperatura mayor que el aire circundante, parte de este calor se transfiere al aire por convección.

EFFECTO DE RETRASO DEL TIEMPO

La energía absorbida por muros, suelo, mobiliario... contribuye a la carga de refrigeración de ese espacio, pero solo después de un retraso en el tiempo. Parte de esta energía está todavía presente y rerradiante incluso después de que las fuentes de calor hayan sido apagadas o eliminadas.

Hay siempre un retraso significativo entre el tiempo en el que una fuente de calor se activa y el punto cuando la energía rerradiada iguala a la que está siendo almacenada instantáneamente.

La carga requerida para el espacio puede ser mucho más baja que la ganancia de calor instantánea que está siendo generada y el pico de carga puede verse afectado significativamente.

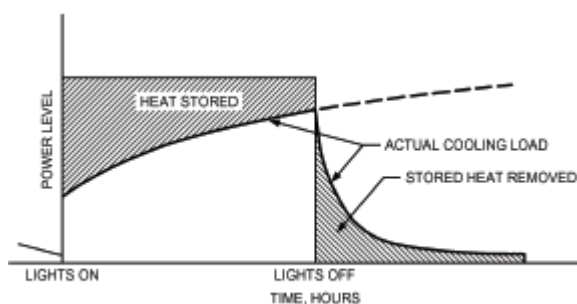


Fig. 2 Thermal Storage Effect in Cooling Load from Lights

En la imagen se puede ver claramente cómo existe un almacenamiento de calor desde que se encienden las luces hasta que se apagan, y cómo va aumentando la carga de refrigeración. En el momento que se apagan las luces, la carga de refrigeración actual se reduce considerablemente, aunque existe todavía un calor almacenado eliminado.

Como ya se ha comentado, se ha utilizado el Radiant Time Series (RTS) Method.

Características

- Cuantifica la contribución de cada componente en la carga de refrigeración total
- Es adecuado para el diseño de cálculos de pico de carga, pero no debe ser usado para simulaciones de energía anual por sus suposiciones.

Suposiciones y principios

- Periodo estacionario
- Existen 2 retrasos de tiempo: retraso de la ganancia de calor conductiva a través de las superficies exteriores opacas y retraso de la ganancia de calor radiante convertida a carga de refrigeración.

Muros exteriores y tejados conducen calor por las diferencias de temperatura entre el aire interior y el exterior. La energía solar en las superficies exteriores es absorbida y más tarde transferida por conducción al interior del edificio. Debido a la masa de las construcciones y a su capacidad térmica, existe un retraso en el tiempo entre la entrada de calor a través de la superficie exterior y la ganancia de calor en el interior.

Muchas fuentes de calor transfieren energía a una habitación por una combinación de convección y radiación:

- La parte convectiva de la ganancia de calor se convierte inmediatamente en carga de refrigeración.
- La parte radiante es absorbida por la masa de las superficies interiores y se convierte en carga de refrigeración cuando es transferida, más tarde, por convección, de esas superficies al aire de la habitación. Así, de este modo, las ganancias de calor radiante llegan a ser cargas de refrigeración después de un periodo de tiempo.

El método RTS considera los retrasos del tiempo por conducción y radiación multiplicando las ganancias de calor de cada hora por unas series de tiempo de 24 horas. Aquí aparecen unos factores:

- Radiant Time Factors (RTF): refleja el porcentaje de una ganancia de calor radiante anterior llega a ser carga de refrigeración durante la hora actual.
- Conduction Time Factors (CTF): refleja el porcentaje de una ganancia de calor anterior (en el exterior de un muro, tejado...) que llega a ser ganancia de calor en el interior durante la hora actual.

PROCEDIMIENTO de RTS

- 1.- Calcula el perfil horario (24 horas) de cada ganancia de calor para el día de diseño.
- 2.- Divide las ganancias de calor en parte radiante y parte convectiva.

Table 14 Recommended Radiative/Convective Splits for Internal Heat Gains

Heat Gain Type	Recommended Radiative Fraction	Recommended Convective Fraction	Comments
Occupants, typical office conditions	0.6	0.4	See Table 1 for other conditions.
Equipment:	0.1 to 0.8	0.9 to 0.2	See Tables 6 to 12 for details of equipment heat gain and recommended radiative/convective splits for motors, cooking appliances, laboratory equipment, medical equipment, office equipment, etc. Varies; see Table 3.
Office, with fan	0.10	0.9	
Without fan	0.3	0.7	
Lighting			
Conduction heat gain			
Through walls and floors	0.46	0.54	
Through roof	0.60	0.40	
Through windows	0.33 (SHGC > 0.5) 0.46 (SHGC < 0.5)	0.67 (SHGC > 0.5) 0.54 (SHGC < 0.5)	
Solar heat gain through fenestration			
Without interior shading	1.0	0.0	Varies; see Tables 13A to 13G in Chapter 15.
With interior shading			
Infiltration	0.0	1.0	

- 3.- Aplica los apropiados RTS a la parte radiante de las ganancias de calor, para tener en cuenta los retrasos en el tiempo.
- 4.- Suma la parte convectiva y la parte radiante “retrasada” de las ganancias para determinar la carga de refrigeración de cada hora y de cada componente.

Después de calcular las cargas de refrigeración de cada componente para cada hora, se suman aquellas que determinan la carga total para cada hora y se selecciona la hora con la carga pico, para poder diseñar los sistemas a utilizar. Habrá que repetir con otros meses.

CÁLCULO DE LA CARGA DE REFRIGERACIÓN

La carga de refrigeración para cada componente (iluminación, ocupación, muros, tejados, equipos...) en una hora particular es la suma de la parte convectiva de la ganancia de calor más la parte radiante (con retraso en el tiempo) para esa hora y las 23 horas previas.

RTS convierte la parte radiante de la ganancia en carga de refrigeración para cada hora usando los Radiant Time Factors.

$$Q_{r,\theta} = r_0 q_{r,\theta} + r_1 q_{r,\theta-1} + r_2 q_{r,\theta-2} + r_3 q_{r,\theta-3} + \dots + r_{23} q_{r,\theta-23}$$

where

$Q_{r,\theta}$ –radiant cooling load Q_r for current hour θ , W

$q_{r,\theta}$ –radiant heat gain for current hour, W

$q_{r,\theta-s}$ –radiant heat gain s hours ago, W

$r_0, r_1, \text{ etc.}$ –radiant time factors

Para ello se utilizan 2 RTS diferentes:

- Solar: para ganancias de calor solar directas. Se asume que la energía radiante es distribuída solo por el suelo y mobiliario.
- Nonsolar: para el resto de ganancias de calor. Se asume que la energía radiante es distribuída uniformemente por todas las superficies internas. Nonsolar RTS se aplica para la ocupación, iluminación, equipos, muros, tejados y suelos. También para ganancias de calor solar difuso y directo de huecos.

El cálculo de las cargas de calefacción es el método típico en el cual se evalúan las pérdidas de calor de los espacios en función de la transmitancia térmica de los cerramientos exteriores, huecos y puentes térmicos. Al igual que en residencial, se reduce a:

$$Q = U \cdot A \cdot \Delta T \quad (\text{W})$$

Si a esto se añaden los cálculos de ventilación e infiltración, ya se tienen las pérdidas de calor.

6.4. TablasASHRAE.py

Este fichero contiene multitud de funciones y todas las tablas que aparecen en los capítulos 17 y 18 del libro Fundamentals 2009 que se hacen absolutamente imprescindibles a la hora de programar.

Para empezar, existen 2 funciones, `def interpola` y `def interpolaSoloX` que me ayudarán a obtener los valores exactos en todas las tablas en las que sea necesario interpolar. La función `interpola`, me da el valor exacto en aquellas tablas en las que haya que interpolar en ambos ejes:

Table 5 Typical IDF Values, L/(s·cm²)

H, m	Heating Design Temperature, °C					Cooling Design Temperature, °C			
	-40	-30	-20	-10	0	10	30	35	40
2.5	0.10	0.095	0.086	0.077	0.069	0.060	0.031	0.035	0.040
3	0.11	0.10	0.093	0.083	0.072	0.061	0.032	0.038	0.043
4	0.14	0.12	0.11	0.093	0.079	0.065	0.034	0.042	0.049
5	0.16	0.14	0.12	0.10	0.086	0.069	0.036	0.046	0.055
6	0.18	0.16	0.14	0.11	0.093	0.072	0.039	0.050	0.061
7	0.20	0.17	0.15	0.12	0.10	0.075	0.041	0.051	0.068
8	0.22	0.19	0.16	0.14	0.11	0.079	0.043	0.058	0.074

mientras que en la función `interpolaSoloX`, el valor en el eje y lo tenemos fijado de antemano.

Table 10 Peak Irradiance, W/m²

Exposure		Latitude								
		20°	25°	30°	35°	40°	45°	50°	55°	60°
North	E_D	125	106	92	84	81	85	96	112	136
	E_d	128	115	103	93	84	76	69	62	55
	E_t	253	221	195	177	166	162	164	174	191
Northeast/Northwest	E_D	460	449	437	425	412	399	386	374	361
	E_d	177	169	162	156	151	147	143	140	137
	E_t	637	618	599	581	563	546	529	513	498
East/West	E_D	530	543	552	558	560	559	555	547	537
	E_d	200	196	193	190	189	188	187	187	187
	E_t	730	739	745	748	749	747	742	734	724
Southeast/Southwest	E_D	282	328	369	405	436	463	485	503	517
	E_d	204	203	203	204	205	207	210	212	215
	E_t	485	531	572	609	641	670	695	715	732
South	E_D	0	60	139	214	283	348	408	464	515
	E_d	166	193	196	200	204	209	214	219	225
	E_t	166	253	335	414	487	557	622	683	740
Horizontal	E_D	845	840	827	806	776	738	691	637	574
	E_d	170	170	170	170	170	170	170	170	170
	E_t	1015	1010	997	976	946	908	861	807	744

A partir de aquí, el fichero se divide en 3 partes:

- Las tablas que son utilizadas por `ASHRAE.py` (para `CE3X-Residencial`)
- Las tablas que son utilizadas por `ASHRAETerciario.py` (para `CE3X- Terciario`)
- La función `temperaturasDiseno`.

Las tablas utilizadas para Residencial son todas sacadas del libro Fundamentals 2009 de la ASHRAE. En el código fuente de `TablasASHRAE.py`, que se encuentra en el apartado 5.1.4., se explica para qué se utiliza cada una y qué parámetros se le tienen que pasar. Si se quiere saber en qué orden las utiliza el programa hay que dirigirse al apartado 5.1.2.:

Para Terciario, las tablas y funciones que se utilizan son las siguientes. En el código fuente se explica el uso que se le da en el programa, en el apartado 5.1.3., mientras que en 5.1.4. los parámetros que se le deben pasar y su razón de ser:

- `perfilUso`,
- `horarioFuncionamiento`,

- actividad,
- SolarRTS
- NonsolarRTS
- wallCTS
- roofCTS
- orientación
- beamshgc

La función temperaturasDiseno contiene la siguiente información para cada provincia:

- Temperatura de refrigeración (percentil 1%)
- Temperatura de calefacción (percentil 99%)
- Oscilación Media Diaria (OMD)
- Array con la temperatura, hora a hora, del día más caluroso de Julio.
- Array con la temperatura, hora a hora, del día más caluroso de Agosto.
- Array con la temperatura, hora a hora, del día más caluroso de Septiembre.

7. CONCLUSIONES

Actualmente está muy de moda hablar del cambio climático, del calentamiento global, del consumo energético, de las energías renovables y como no de la eficiencia energética.

Cuando se habla de eficiencia energética, comúnmente se piensa en ahorro de electricidad. Pero la definición de eficiencia y de energía desde una perspectiva ecológica global involucra mucho más que eso.

La eficiencia -que dicho sea de paso es muy distinta a la eficacia-, es la relación entre los recursos invertidos en realizar una acción y el producto obtenido de esta. A menor cantidad de recursos invertidos para lograr el mismo objetivo, mayor es la eficiencia de esa acción, proceso o sistema. La energía, por su parte, se define de distintas maneras según la disciplina desde la que se aborda, pero todas estas acepciones parten de un origen griego común; *energeia*, actividad, operación y *energós*, fuerza de acción.

Tenemos entonces, que ser eficientes en el uso de la energía significa utilizar la menor fuerza de acción (energía) para conseguir el mayor objetivo (eficiencia), logrando así una relación óptima (que no puede ser mejor).

La eficiencia energética es un combate al derroche poniendo atención al ciclo completo de un producto o una acción, promoviendo el uso inteligente de los recursos en un mundo finito, de recursos limitados. Y un escenario clave para acabar con dicho derroche es el parque edificatorio mundial, un sector que se lleva gran parte de los recursos.

Ello requiere, de un lado, definir una estrategia, y, de otro, establecer un amplio acuerdo entre todos los sectores afectados: constructores y promotores, fabricantes y asociaciones de productos de la construcción, las administraciones públicas – Administración General del Estado, comunidades autónomas y administraciones locales–, consumidores y usuarios, sindicatos y ONG, arquitectos y profesionales, universidades y centros de I+D+i.

Se deduce, por tanto, que es necesario cumplir con tres condiciones básicas: la autocontención en la evolución del parque edificatorio; la ecoeficiencia, a aplicar tanto en la nueva edificación como en el stock edificatorio existente; y la complicidad de todos los actores implicados.

Mediante la combinación de las tres condiciones y las lógicas de sostenibilidad, es posible reconducir de forma más satisfactoria y a medio plazo, el comportamiento ambiental de la edificación.

En breve se regulará la certificación energética de edificios existentes para cumplir por completo con las exigencias de la Directiva Europea en materia de certificación energética de edificios. Este futuro Real Decreto obligará a que todos los edificios existentes, cuando se vendan o alquilen, dispongan de un certificado de eficiencia energética, independientemente de que sufran o no modificaciones, reformas o rehabilitaciones.

Con esta nueva normativa surgirá la necesidad de disponer de herramientas adecuadas que permitan calificar un gran volumen de edificios y viviendas existentes, y es ahí donde aparece CE3X, una herramienta fiable y rápida que permite calificar edificios y viviendas existentes, y que está en un proceso de mejora continua para satisfacer las necesidades de los usuarios y del nuevo marco normativo.

Además, con la implantación del plugin se conseguirá averiguar qué necesidades energéticas tienen los distintos edificios y lo que es más importante, saber si un edificio tiene cubiertas dichas necesidades, ya que puede ocurrir que un edificio tenga una alta eficiencia energética (tenga por ejemplo una calificación A) pero a costa de tener unas calderas que son insuficientes para la cantidad de superficie a la que satisfacen. Gracias a este complemento se puede saber si las instalaciones están correctamente dimensionadas en una cierta ciudad.

8. MANUAL DE USUARIO

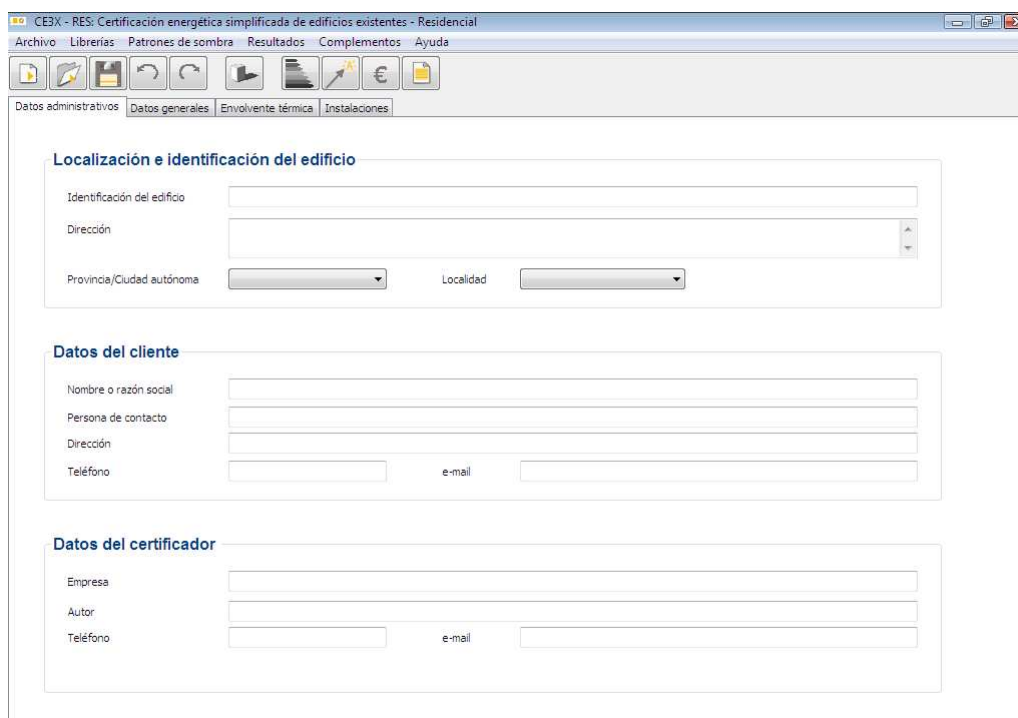
En este apartado se narrará cómo hacer uso del plugin que se explica en este Trabajo Técnico.

En primer lugar, cuando el usuario hace doble clic sobre el icono, le aparecerá en la pantalla la siguiente ventana de entrada al programa de “Certificación Energética Simplificada de Edificios Existentes”. El usuario deberá elegir entre tres opciones diferentes según el tipo de edificio a certificar. Deberá pinchar en el botón Residencial cuando se certifique un bloque de viviendas, un unifamiliar o una vivienda individual.

Cuando se proceda a certificar un edificio del sector terciario se pinchará en “Pequeño Terciario” o “Gran Terciario”.



La siguiente ventana es la que aparece una vez que se está dentro de CE³X. En este caso, como se ve en la barra de título, se está en Residencial.



CE3X - RES: Certificación energética simplificada de edificios existentes - Residencial

Archivo Librerías Patrones de sombra Resultados Complementos Ayuda

Datos administrativos **Datos generales** Envolverte térmica Instalaciones

Localización e identificación del edificio

Identificación del edificio

Dirección

Provincia/Ciudad autónoma Localidad

Datos del cliente

Nombre o razón social

Persona de contacto

Dirección

Teléfono e-mail

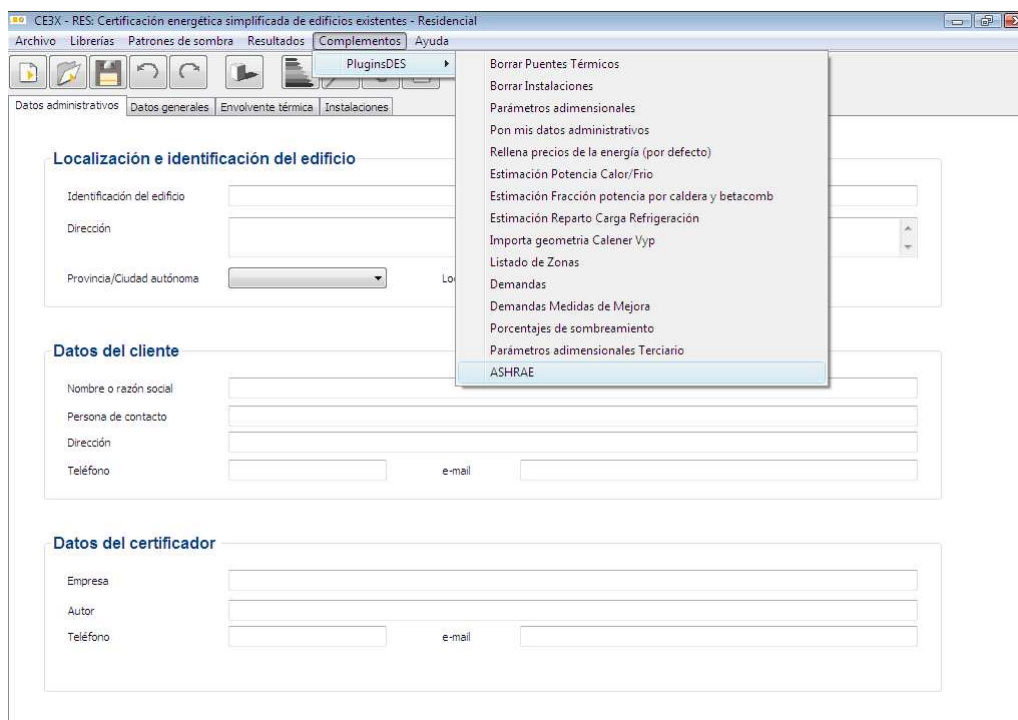
Datos del certificador

Empresa

Autor

Teléfono e-mail

Para acceder al plugin, se hace clic en *Complementos*, dentro del Menú principal y aparecen todos y cada uno de los plugins que posee el programa. El plugin que aquí se explica está en la última posición de todos y se llama, ASHRAE. En el Manual de usuario del programa, se explica que se obtiene con este complemento.



CE3X - RES: Certificación energética simplificada de edificios existentes - Residencial

Archivo Librerías Patrones de sombra Resultados **Complementos** Ayuda

Datos administrativos Datos generales Envolverte térmica Instalaciones

Localización e identificación del edificio

Identificación del edificio

Dirección

Provincia/Ciudad autónoma Localidad

Datos del cliente

Nombre o razón social

Persona de contacto

Dirección

Teléfono e-mail

Datos del certificador

Empresa

Autor

Teléfono e-mail

PluginsDES

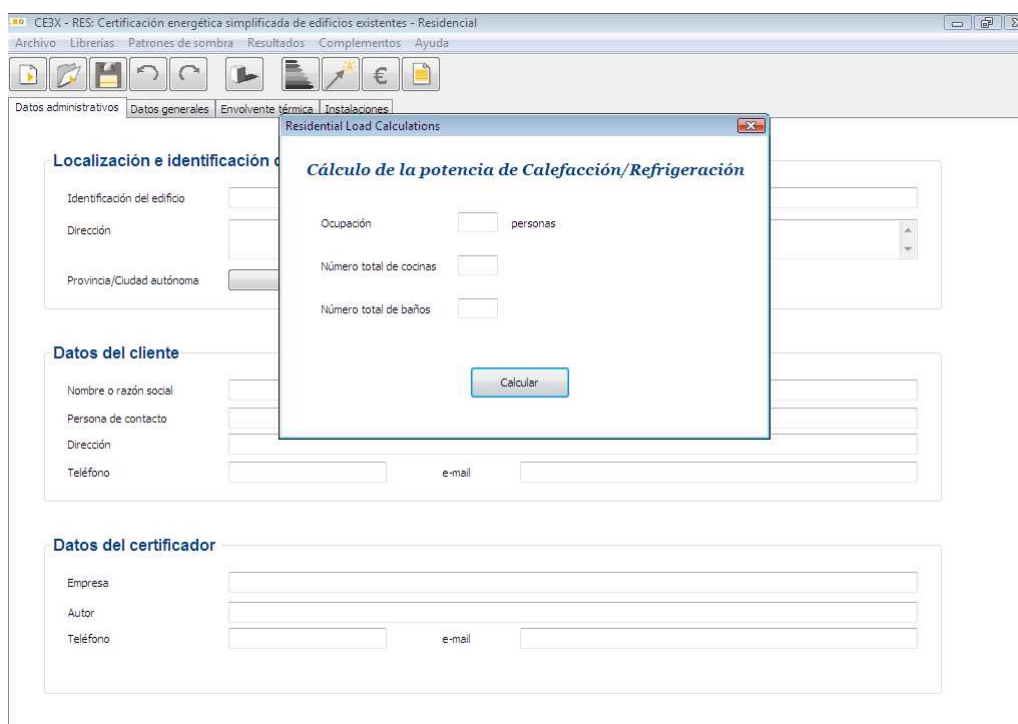
- Borrar Puentes Térmicos
- Borrar Instalaciones
- Parámetros adimensionales
- Pon mis datos administrativos
- Rellena precios de la energía (por defecto)
- Estimación Potencia Calor/Frío
- Estimación Fracción potencia por caldera y betacomb
- Estimación Reparto Carga Refrigeración
- Importa geometría Calener Vyp
- Listado de Zonas
- Demandas
- Demandas Medidas de Mejora
- Porcentajes de sombreado
- Parámetros adimensionales Terciario
- ASHRAE**

Cuando se hace clic sobre ASHRAE, aparecerá una ventana, que será diferente según el programa en que nos encontremos. La imagen siguiente se corresponde con el programa CE³X-Residencial, como se puede ver en la barra de título, “Residential Load Calculations”, que precisamente es el nombre del capítulo 17 del libro Fundamentals 2009.

Los campos que pide el programa al usuario para la realización de los cálculos son:

- Ocupación: número de personas que habitualmente habitan el edificio.
- Número total de cocinas
- Número total de baños

Estos dos últimos campos son fundamentales para el cálculo de la ventilación.



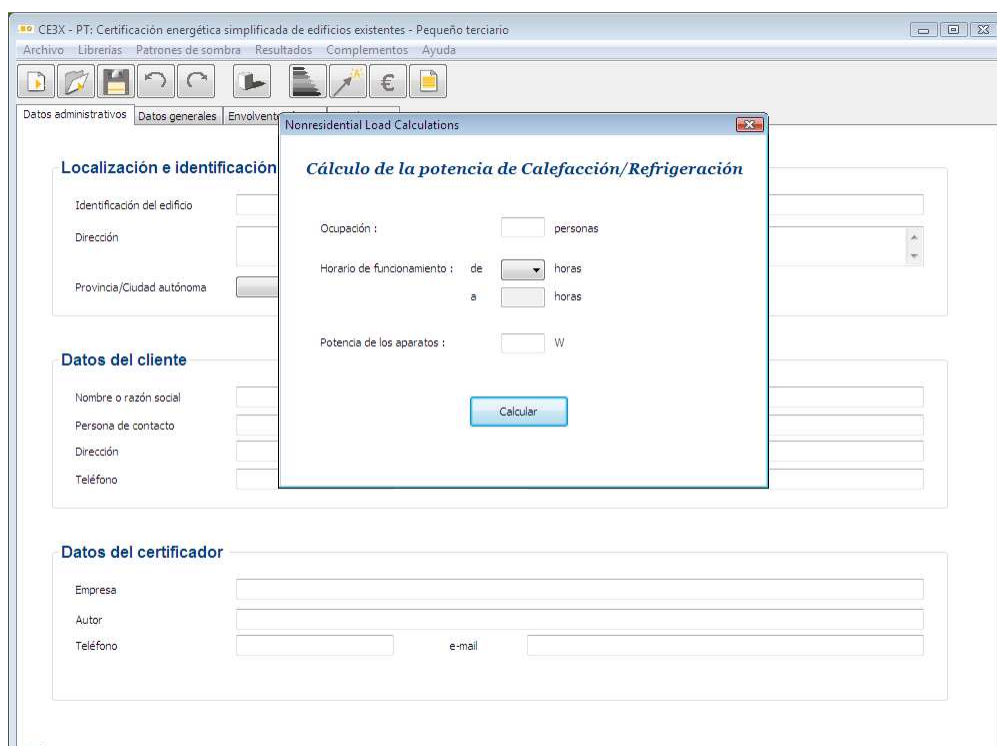
Por otra parte, para terciario, la ventana es la siguiente. En la barra de título aparece el nombre del capítulo 18 del libro Fundamentals 2009, “Nonresidential Load Calculations”.

Los campos que se piden ahora son:

- Ocupación: igual que el anterior.
- Horario de funcionamiento: el horario que tiene la mayoría de los ocupantes. La hora de finalización no es posible modificarla, ya que viene impuesta por la hora de inicio y por el perfil de uso (que se escoge en la pestaña, Datos Generales). Si se ha escogido “Intensidad Media-12h”, y la hora de inicio son las 6 de la

mañana, la hora de finalización serán las 18 horas y así respectivamente con los diferentes perfiles de uso.

- Potencia de los aparatos: es la potencia, en Watios, de los aparatos existentes en el edificio, ya sean ordenadores, impresoras, pantallas, así como hornos, microondas...



The screenshot shows the CE3X software interface for simplified energy certification of existing buildings. A dialog box titled 'Nonresidential Load Calculations' is open, displaying the 'Cálculo de la potencia de Calefacción/Refrigeración' (Calculation of heating/cooling power) section. The dialog includes input fields for 'Ocupación' (Occupancy) in persons, 'Horario de funcionamiento' (Operating hours) from 'de' to 'a' in hours, and 'Potencia de los aparatos' (Appliance power) in Watts (W). A 'Calcular' (Calculate) button is at the bottom of the dialog. The background interface shows tabs for 'Datos administrativos', 'Datos generales', and 'Envoltorio', with 'Datos generales' currently selected. It also displays sections for 'Localización e identificación', 'Datos del cliente', and 'Datos del certificador'.

9. MANUAL DE PROGRAMADOR

9.1. LENGUAJE DE PROGRAMACIÓN

Un lenguaje de programación es un idioma artificial diseñado para expresar computaciones que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación.

También la palabra programación se define como el proceso de creación de un programa de computadora, mediante la aplicación de procedimientos lógicos, a través de los siguientes pasos:

- El desarrollo lógico del programa para resolver un problema en particular.
- Escritura de la lógica del programa empleando un lenguaje de programación específico (codificación del programa).
- Ensamblaje o compilación del programa hasta convertirlo en lenguaje de máquina.
- Prueba y depuración del programa.
- Desarrollo de la documentación.
- Existe un error común que trata por sinónimos los términos 'lenguaje de programación' y 'lenguaje informático'. Los lenguajes informáticos engloban a los lenguajes de programación y a otros más, como por ejemplo HTML (lenguaje para el marcado de páginas web que no es propiamente un lenguaje de programación, sino un conjunto de instrucciones que permiten diseñar el contenido de los documentos).
- Permite especificar de manera precisa sobre qué datos debe operar una computadora, cómo deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural. Una característica relevante de los lenguajes de programación es precisamente que más de un programador pueda usar un conjunto común de instrucciones que sean comprendidas entre ellos para realizar la construcción de un programa de forma colaborativa.

Historia

Para que la computadora entienda nuestras instrucciones debe usarse un lenguaje específico conocido como código máquina, el cual la máquina comprende fácilmente, pero que lo hace excesivamente complicado para las personas. De hecho sólo consiste en cadenas extensas de números 0 y 1.

Para facilitar el trabajo, los primeros operadores de computadoras decidieron hacer un traductor para reemplazar los 0 y 1 por palabras o abstracción de palabras y letras provenientes del inglés; éste se conoce como lenguaje ensamblador. Por ejemplo, para sumar se usa la letra A de la palabra inglesa add (sumar). El lenguaje ensamblador sigue la misma estructura del lenguaje máquina, pero las letras y palabras son más fáciles de recordar y entender que los números.

La necesidad de recordar secuencias de programación para las acciones usuales llevó a denominarlas con nombres fáciles de memorizar y asociar: ADD (sumar), SUB (restar), MUL (multiplicar), CALL (ejecutar subrutina), etc. A esta secuencia de posiciones se le denominó "instrucciones", y a este conjunto de instrucciones se le llamó lenguaje ensamblador. Posteriormente aparecieron diferentes lenguajes de programación, los cuales reciben su denominación porque tienen una estructura sintáctica similar a los lenguajes escritos por los humanos, denominados también lenguajes de alto nivel.

Elementos

Todos los lenguajes de programación tienen algunos elementos de formación primitivos para la descripción de los datos y de los procesos o transformaciones aplicadas a estos datos (tal como la suma de dos números o la selección de un elemento que forma parte de una colección). Estos elementos primitivos son definidos por reglas sintácticas y semánticas que describen su estructura y significado respectivamente.

Sintaxis

```
def add5(x):
    return x+5

def dotwrite(ast):
    nodename = getnodename()
    label=symbol.syn name.get(int(ast[0]).ast[0])
    print ' %s [label="%s" % (nodename, label)
    if isinstance(ast[1].str):
        if ast[1].strip():
            print ' "%s":' % ast[1]
        else:
            print ']'
    else:
        print '[':
        children = []
        for i in n. childrenumaxata(ast[1:]):
            children.append(dotwrite(child))
        print ' %s -> [' % nodename
        for i in namechildren
            print ' "%s' % name.
```

Con frecuencia se resaltan los elementos de la sintaxis con colores diferentes para facilitar su lectura. Este ejemplo está escrito en Python.

A la forma visible de un lenguaje de programación se le conoce como sintaxis. La mayoría de los lenguajes de programación son puramente textuales, es decir, utilizan secuencias de texto que incluyen palabras, números y puntuación, de manera similar a los lenguajes naturales escritos. Por otra parte, hay algunos lenguajes de programación que son más gráficos en su naturaleza, utilizando relaciones visuales entre símbolos para especificar un programa.

La sintaxis de un lenguaje de programación describe las combinaciones posibles de los símbolos que forman un programa sintácticamente correcto. El significado que se le da a una combinación de símbolos es manejado por su semántica (ya sea formal o como parte del código duro de la referencia de implementación).

No todos los programas sintácticamente correctos son semánticamente correctos. Muchos programas sintácticamente correctos tienen inconsistencias con las reglas del lenguaje; y pueden (dependiendo de la especificación del lenguaje y la solidez de la implementación) resultar en un error de traducción o ejecución. En algunos casos, tales programas pueden exhibir un comportamiento indefinido. Además, incluso cuando un programa está bien definido dentro de un lenguaje, todavía puede tener un significado que no es el que la persona que lo escribió estaba tratando de construir.

Usando el lenguaje natural, por ejemplo, puede no ser posible asignarle significado a una oración gramáticamente válida o la oración puede ser falsa:

- "Las ideas verdes y descoloridas duermen furiosamente" es una oración bien formada gramáticamente pero no tiene significado comúnmente aceptado.
- "Juan es un soltero casado" también está bien formada gramáticamente pero expresa un significado que no puede ser verdadero.

9.2. PYTHON

Python  python TM	
<u>Extensiones comunes:</u>	.py .pyc .pyd .pyo .pyw
<u>Paradigma:</u>	multiparadigma : orientado a objetos , imperativo , funcional , reflexivo
<u>Apareció en:</u>	1991
<u>Diseñado por:</u>	Guido van Rossum
<u>Desarrollador:</u>	Python Software Foundation
<u>Última versión estable:</u>	3.1.2 / 2.7 (21 de marzo de 2010 / 03 de julio de 2010)
<u>Tipo de dato:</u>	fuertemente tipado , dinámico
<u>Implementaciones:</u>	CPython , IronPython , Jython , Python for S60, PyPy , Unladen Swallow
<u>Dialectos:</u>	Stackless Python , RPython
<u>Influido por:</u>	ABC , ALGOL 68 , C , Haskell , Icon , Lisp , Modula-3 , Perl , Smalltalk , Java
<u>Ha influido a:</u>	Boo , Cobra , D , Falcon , Genie , Groovy , Ruby , JavaScript
<u>Sistema operativo:</u>	Multiplataforma
<u>Licencia de software:</u>	Python Software Foundation License
<u>Web:</u>	http://www.python.org/

Python es un lenguaje de programación de alto nivel cuya filosofía hace hincapié en una sintaxis muy limpia y que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico, es fuertemente tipado y es multiplataforma.

Lenguaje interpretado o de script

Un lenguaje interpretado o de script es aquel que se ejecuta utilizando un programa intermedio llamado intérprete, en lugar de compilar el código a lenguaje máquina que pueda comprender y ejecutar directamente una computadora (lenguajes compilados).

La ventaja de los lenguajes compilados es que su ejecución es más rápida. Sin embargo los lenguajes interpretados son más flexibles y más portables.

Python tiene, no obstante, muchas de las características de los lenguajes compilados, por lo que se podría decir que es semi interpretado. En Python, como en Java y muchos otros lenguajes, el código fuente se traduce a un pseudo código máquina intermedio llamado bytecode la primera vez que se ejecuta, generando archivos .pyc o .pyo (bytecode optimizado), que son los que se ejecutarán en sucesivas ocasiones.

Tipado dinámico

La característica de tipado dinámico se refiere a que no es necesario declarar el tipo de dato que va a contener una determinada variable, sino que su tipo se determinará en tiempo de ejecución según el tipo del valor al que se asigne, y el tipo de esta variable puede cambiar si se le asigna un valor de otro tipo.

Fuertemente tipado

No se permite tratar a una variable como si fuera de un tipo distinto al que tiene, es necesario convertir de forma explícita dicha variable al nuevo tipo previamente. Por ejemplo, si tenemos una variable que contiene un texto (variable de tipo cadena o string) no podremos tratarla como un número (sumar la cadena “9” y el número 8). En otros lenguajes el tipo de la variable cambiaría para adaptarse al comportamiento esperado, aunque esto es más propenso a errores.

Multiplataforma

El intérprete de Python está disponible en multitud de plataformas (UNIX, Solaris, Linux, DOS, Windows, OS/2, Mac OS, etc.) por lo que si no utilizamos librerías específicas de cada plataforma nuestro programa podrá correr en todos estos sistemas sin grandes cambios.

Orientado a objetos

La orientación a objetos es un paradigma de programación en el que los conceptos del mundo real relevantes para nuestro problema se trasladan a clases y objetos en nuestro programa. La ejecución del programa consiste en una serie de interacciones entre los objetos.

Historia

Python fue creado a finales de los ochenta por Guido van Rossum en el National Research Institute for Mathematics and Computer Science (CWI), en los Países Bajos, como un sucesor del lenguaje de programación ABC.

Características y paradigmas

Python es un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones.

Python usa tipado dinámico y conteo de referencias para la administración de memoria. Una característica importante de Python es la resolución dinámica de nombres; es decir, lo que enlaza un método y un nombre de variable durante la ejecución del programa (también llamado ligadura dinámica de métodos).

Otro objetivo del diseño del lenguaje es la facilidad de extensión. Se pueden escribir nuevos módulos fácilmente en C o C++. Python puede incluirse en aplicaciones que necesitan una interfaz programable.

Filosofía

Los usuarios de Python se refieren a menudo a la Filosofía Python que es bastante análoga a la filosofía de Unix. El código que sigue los principios de Python de legibilidad y transparencia se dice que es "pythonico". Contrariamente, el código opaco u ofuscado es bautizado como "no pythonico" ("unpythonic" en inglés). Estos principios fueron:

- Bello es mejor que feo.
- Explícito es mejor que implícito.
- Simple es mejor que complejo.
- Complejo es mejor que complicado.
- Plano es mejor que anidado.
- Ralo es mejor que denso.
- La legibilidad cuenta.
- Los casos especiales no son tan especiales como para quebrantar las reglas.
- Aunque lo práctico gana a la pureza.
- Los errores nunca deberían dejarse pasar silenciosamente.
- A menos que hayan sido silenciados explícitamente.
- Frente a la ambigüedad, rechaza la tentación de adivinar.
- Debería haber una -y preferiblemente sólo una- manera obvia de hacerlo.
- Aunque esa manera puede no ser obvia al principio a menos que usted sea holandés.
- Ahora es mejor que nunca.
- Aunque *nunca* es a menudo mejor que *ya mismo*.
- Si la implementación es difícil de explicar, es una mala idea.
- Si la implementación es fácil de explicar, puede que sea una buena idea.

Modo interactivo

El intérprete de Python estándar incluye un *modo interactivo*, en el cual se escriben las instrucciones en una especie de intérprete de comandos: las expresiones pueden ser introducidas una a una, pudiendo verse el resultado de su evaluación inmediatamente. Esto resulta útil tanto para las personas que se están familiarizando con el lenguaje como también para los programadores más avanzados: se pueden probar porciones de código en el modo interactivo antes de integrarlo como parte de un programa.

Existen otros programas, tales como IDLE, bpython o IPython, que añaden funcionalidades extra al modo interactivo, como el auto-completado de código y el coloreado de la sintaxis del lenguaje.

Ejemplo del modo interactivo:

```
>>> 1+1
2

>>> a = range(10)
>>> print a
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Elementos del lenguaje

Python fue diseñado para ser leído con facilidad. Entre otras cosas se utilizan palabras en inglés donde otros lenguajes utilizarían símbolos (por ejemplo, los operadores lógicos `!`, `||` y `&&` en Python se escriben `not`, `or` y `and`, respectivamente).

El contenido de los bloques de código (bucles, funciones, clases, etc.) es declarado mediante espacios o tabuladores (indentación) antes de cada línea de órdenes pertenecientes al bloque. Python se diferencia así de otros muchos lenguajes de programación que mantienen como costumbre declarar los bloques mediante un conjunto de caracteres, normalmente entre llaves `{ }`. Aunque no en todos es así, por ejemplo, el lenguaje de programación Ruby sigue el mismo sistema para declarar los bloques que Python. Se recomienda que la indentación sea siempre de espacios o siempre de tabuladores, pero no mezclarlos, utilizar espacios en vez de tabuladores y que estos siempre sean 4 espacios.

Función factorial en C (indentación opcional)

```
int factorial(int x) {
    if (x == 0)
        return 1;
    else
        return x * factorial(x - 1);
}
```

Función factorial en Python (indentación obligatoria)

```
def factorial(x):
    if x == 0:
        return 1
    else:
        return x * factorial(x - 1)
```

Comentarios

Los *comentarios* se inician con el símbolo #, y se extienden hasta el final de la línea. El intérprete no tiene en cuenta los *comentarios*, lo cual es útil si deseamos poner información adicional en nuestro código como, por ejemplo, una explicación sobre el comportamiento de una sección del programa.

Comentario en una línea en Python

print "Hola mundo" *# También es posible añadir un comentario al final de una línea de código*

Variables

Las *variables* se definen de forma dinámica. Se usa el símbolo = para asignar valores.

x = 1

x = "texto" *# Esto es posible porque los tipos son asignados dinámicamente*

Tipos de datos

Tipo	Clase	Notas	Ejemplo
str	Cadena	Inmutable	'Cadena'
unicode	Cadena	Versión Unicode de str	u'Cadena'
list	Secuencia	Mutable, puede contener objetos de diversos tipos	[4.0, 'Cadena', True]
tuple	Secuencia	Inmutable, puede contener objetos de diversos tipos	(4.0, 'Cadena', True)
set	Conjunto	Mutable, sin orden, no contiene duplicados	set([4.0, 'Cadena', True])
frozenset	Conjunto	Inmutable, sin orden, no contiene duplicados	frozenset([4.0, 'Cadena', True])
dict	Mapping	Grupo de pares clave:valor	{'key1': 1.0, 'key2': False}
int	Número entero	Precisión fija, convertido en long en caso de overflow.	42
long	Número entero	Precisión arbitraria	42L ó 456966786151987643L
float	Número decimal	Coma flotante de doble precisión	3.1415927
bool	Booleano	Valor booleano verdadero o falso	True o False

9.3. ENTORNO DE DESARROLLO INTEGRADO

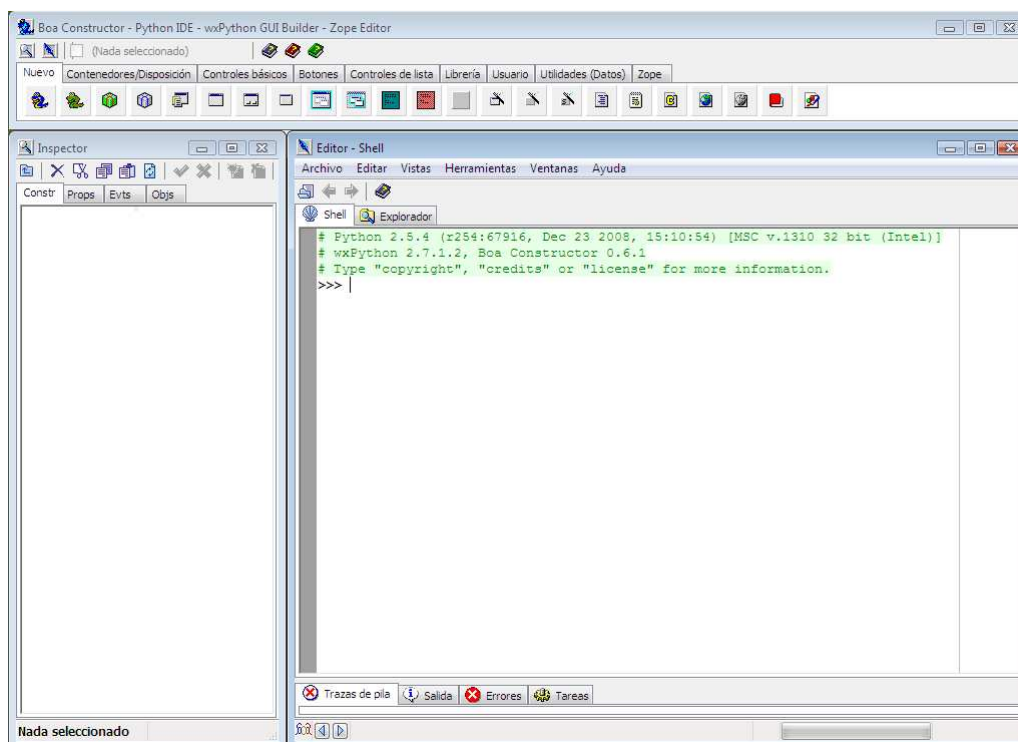
Para poder programar es necesario un entorno de desarrollo integrado, en inglés IDE (Integrated Developed Environment), que no es otra cosa que un programa informático compuesto por un conjunto de herramientas de programación.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI).

Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, **Python**, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva.

Componentes

- Un editor de texto
- Un compilador
- Un intérprete
- Herramientas de automatización
- Un depurador
- Factibilidad para ayudar en la construcción de interfaces gráficas de usuario.



Un **constructor de interfaz gráfica** es una herramienta de programación que simplifica la creación de interfaces gráficas de usuario permitiendo al diseñador organizar widgets utilizando un **editor drag-and-drop WYSIWYG**. Sin un constructor, la interfaz debe ser creada manualmente especificando cada parámetro de los widgets en el código, sin poder observarlo visualmente hasta que el programa esté funcionando.

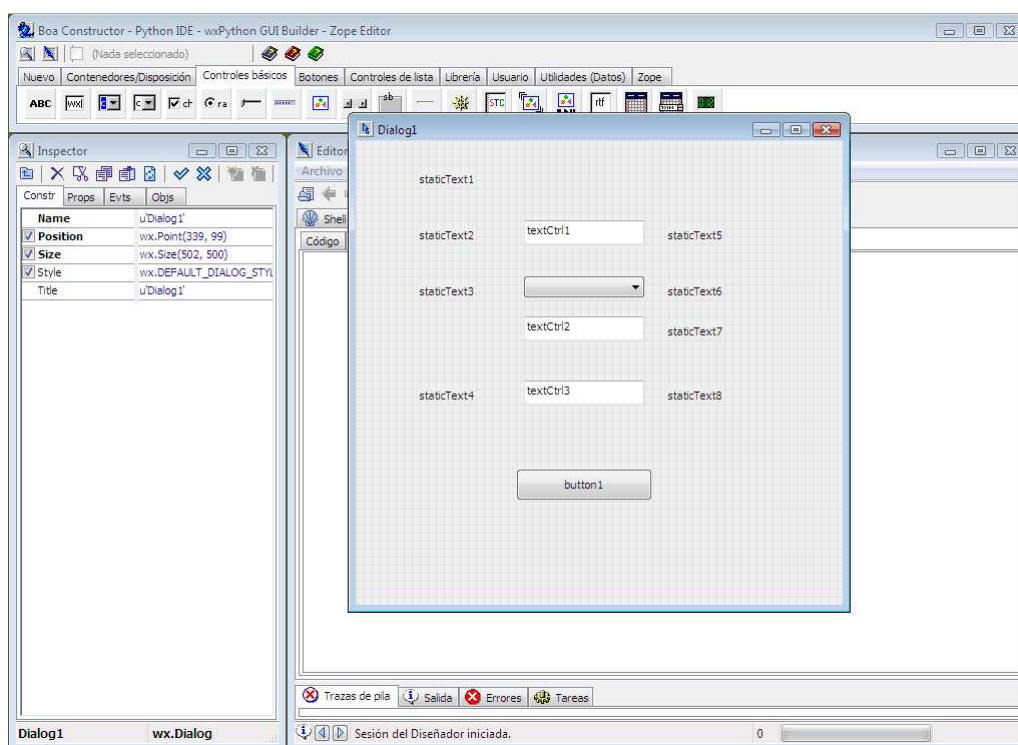
Arrastrar y soltar (*drag and drop*) es una expresión informática que se refiere a la acción de mover con el ratón objetos de una ventana a otra o entre partes de una misma ventana. Los objetos arrastrados son habitualmente archivos, pero también pueden ser arrastrados otros tipos de elementos en función del programa.

Un ejemplo es arrastrar y soltar entre ventanas de un gestor de archivos para cambiarlos de carpeta, o archivos de audio en un reproductor para moverlos de su lista de reproducción.

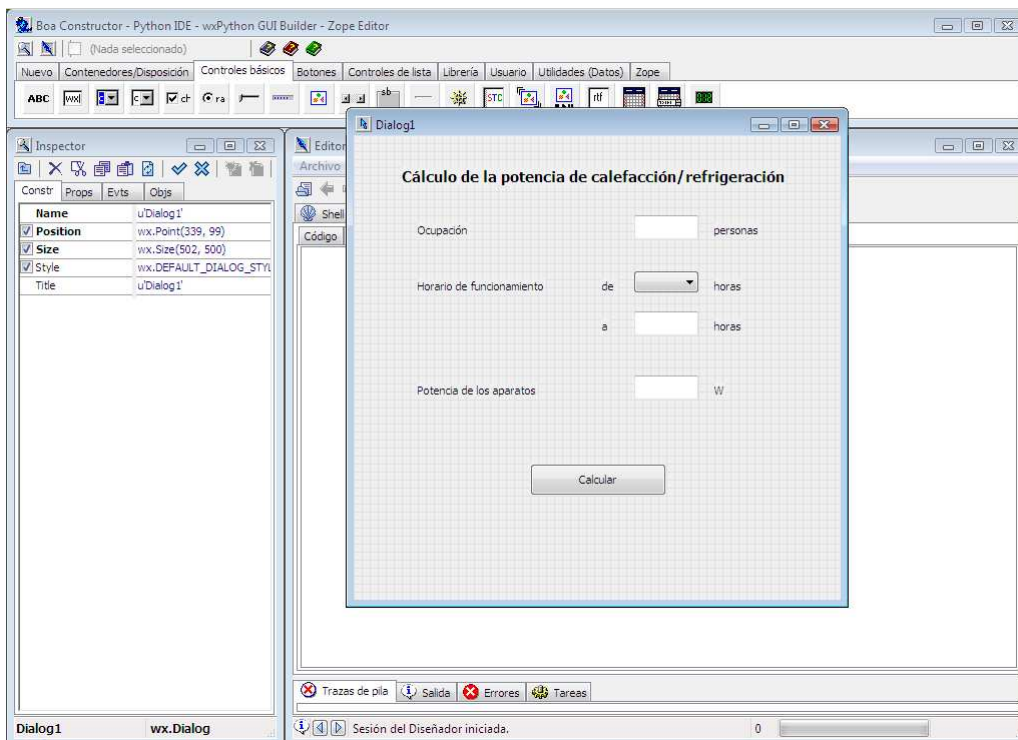
WYSIWYG es el acrónimo de *What You See Is What You Get* (en inglés, "lo que ves es lo que obtienes"). Se aplica a los procesadores de texto y otros editores de texto con formato (como los editores de HTML) que permiten escribir un documento viendo directamente el resultado final, frecuentemente el resultado impreso

En la siguiente imagen se puede observar cómo se realiza la ventana que aparece en el programa de Gran Terciario. Para ello se ha utilizado el Boa Constructor, un entorno de desarrollo integrado (IDE) para python. Arrastrando y soltando, se van colocando todos los objetos que deben aparecer en la ventana. Las opciones que ofrece el editor son muchas y muy variadas. Las que se eligieron en este caso fueron:

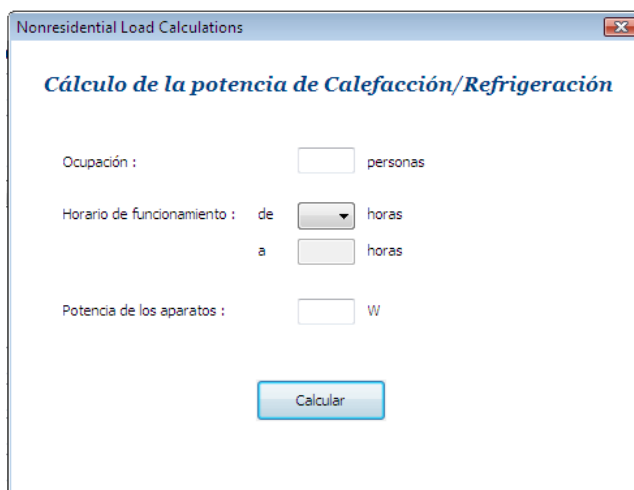
- StaticText: simplemente es el texto que aparece
- TextCtrl: cuadros de texto para que el usuario introduzca datos.
- Choice: desplegables
- Buttons: botones



Los anteriores StaticText, TextCtrl, Choice y Button se van modificando para obtener la ventana final de la forma deseada. Se puede cambiar el tamaño de todos los objetos, modificar su fuente y organizarlos de cualquier manera.



Esta última imagen se corresponde con la ventana que aparece actualmente en el programa.



La **interfaz gráfica de usuario**, conocida también como **GUI** (del inglés *graphical user interface*) es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

10. BIBLIOGRAFÍA

Para la realización de este Trabajo técnico se ha consultado la siguiente bibliografía:

10.1. LIBROS

- TRANSFERENCIA DE CALOR. Yunus A. Çengel. Ed. Mc Graw Hill
- ENCICLOPEDIA DE LA CLIMATIZACIÓN. TOMO I – CALEFACCIÓN. Martín Llorens. Grupo Editorial Ceac. Barcelona 1994
- MANUAL DE CLIMATIZACIÓN. E. Torrella, J. Navarro, R. Cabello y F. Gómez. AMV Ediciones
- Guía Técnica: Condiciones climáticas exteriores de proyecto
Ministerio de industria, turismo y comercio. IDEA

10.2. NORMAS

- Directiva 2002/91/CE del Parlamento Europeo y del Consejo de 16 de diciembre de 2002, relativa a la eficiencia energética de los edificios.
- Real Decreto 47/2007, de 19 de enero, por el que se aprueba el Procedimiento básico para la certificación de eficiencia energética de edificios de nueva construcción.
- Código Técnico de la Edificación, CTE.
- Reglamento de Instalaciones Térmicas en los Edificios (RITE) de 2007.

10.3. PÁGINAS WEB

- www.codigotecnico.org
- www.python.org
- www.wxpython.org
- www.ashrae.org
- www.terra.org
- www.mityc.es
- www.certificacionedificiosexistentes.org
- www.miyabi.es
- www.idae.es

11. ANEXOS

Se adjuntan, a continuación, los capítulos 17 y 18 del libro Fundamentals 2009 de la ASHRAE.

Además, en soporte informático, se adjuntan los siguientes documentos/ficheros:

- ASHRAE.py
- ASHRAETerciario.py
- TablasASHRAE.py
- PluginDES.py
- Guía Técnica: Condiciones climáticas exteriores de proyecto
Ministerio de industria, turismo y comercio. IDAE
- Meteorología: carpeta con ficheros de las distintas ciudades españolas con las temperaturas, hora a hora, de todos los días del año.
- Directiva 2002/91/CE Del Parlamento Europeo
- Real Decreto 47/2007
- ITE 03
- Trabajo Técnico



Firma del autor del proyecto

Pamplona, 19 de Noviembre de 2010